

2002:DS15

EXAMENSARBETE

Utredning av kommunikation inom Skaraborgs Städ AB

-samt utveckling av T.I.S – Trasan Information System

**Caroline Abrahamsson
Viktoria Granli
Åsa Sandberg**

2002-05-24

**Högskolan Trollhättan/Uddevalla
Institutionen för informatik och matematik
Box 957, 461 29 Trollhättan
Tel: 0520-47 53 30 Fax: 0520-47 53 99**

EXAMENSARBETE

Utredning av kommunikation inom Skaraborgs Städ AB – utveckling av T.I.S – Trasan Information System

Sammanfattning

Vi har undersökt hur man kan underlätta kommunikationen mellan personalen på de fyra olika kontoren som ingår i Skaraborgs Städ AB.

Syftet med rapporten är att visa vilka metoder vi använt och hur vi genomfört undersökningarna, den ska även visa hur vi skapat en lösning på det problem vi analyserade fram.

Valet av utvecklingsverktyg blev scriptspråket PHP och databashanteraren MySQL. Anledningen till dessa val är att stöd finns på webbhotellet där uppdragsgivarens nuvarande hemsida ligger.

Resultatet blev en dynamisk webbapplikation efter personalens behov, och önskemål. Applikationen har vi döpt till T.I.S - Trasan Information System. I systemet ingår ett filarkiv med uppladdningsmöjlighet och ett meddelandesystem där man kan göra inlägg och på så vis underlätta kommunikationen mellan de olika kontoren. Vi har skapat ett antal sidor där administratören ska kunna sköta underhållet av systemet på ett enkelt sätt.

Med applikationen vi utvecklat kan personalen undvika onödiga telefonsamtal, stora utskrifter och massutskick av protokoll och nyhetsbrev. Med detta hoppas vi att uppdragsgivarna kan spara in en del dubbelarbete och förenkla sin kommunikation.

Nyckelord: PHP, MySQL, Webbapplikation

Utgivare: Högskolan Trollhättan/Uddevalla, institutionen för informatik och matematik
Box 957, 461 29 Trollhättan
Tel: 0520-47 53 30 Fax: 0520-47 53 99

Författare: Caroline Abrahamsson, Viktoria Granli & Åsa Sandberg

Examinator: Stefan Mankefors, docent och universitetslektor vid HTU

Handledare: Klara Horvath Nagy, HTU

Poäng: 10 **Nivå:** C

Huvudämne: Datavetenskap **Inriktning:** Programmering

Språk: Svenska **Nummer:** 2002:DS15 **Datum:** 2002-05-24

Thesis

Investigation of Communication within Skaraborgs Stöd AB - Development of T.I.S – Trasan Information System

Summary

Our primary goal has been to investigate how to simplify the communication between the employees on the four different offices that Skaraborgs Stöd AB consists of.

The purpose of the report is to show which methods we have used and how we concluded our investigation. The report should also show how we created a solution on the analysed problem.

We choosed PHP as programming language and MySQL as database manager. One of the reasons for our choice was that there was support for both of them on the web host that the company already use.

The result became a dynamic web application, design and developed according to the employees desire and needs. The name of the application is T.I.S – Trasan Information System. The application contains a file archive and a message board. The archive offers the users to upload and download files. In the message board users can write their own message and read others to simplify the communication.

The web application will help the employees to avoid unnecessary phone calls and a lot of paper handling. We hope this will simplify their communication and duplication of work.

Keywords: PHP, MySQL, Web application

Publisher: University of Trollhättan/Uddevalla, Department of informatics and mathematics
Box 957, S-461 29 Trollhättan, SWEDEN
Phone: + 46 520 47 53 30 Fax: + 46 520 47 53 99

Author: Caroline Abrahamsson, Viktoria Granli & Åsa Sandberg

Examiner: Senior Master Stefan Mankefors, HTU

Advisor: Klara Horvath Nagy, HTU

Subject: Computer Science

Language: Swedish **Number:** 2002:DS15 **Date:** 24 May. 02

Förord

Vi vill tacka Skaraborgs Stöd AB för att de varit så engagerade och hjälpsamma under vårt arbete. Vi vill också passa på att ge ett stort tack till Magnus Jonsson, systemarkitekt på företaget Exsense, som har hjälpt, stöttat och givit oss användbara tips när vi kodat. Han har också gett oss tillgång till ramverket GAIA som förenklat och effektiviserat vårt arbete.

Övriga personer som inte får glömmas bort är Viktorias föräldrar som vi vill tacka för kost och logi när vi besökt uppdragsgivarna och Klara Horvath Nagy som varit vår handledare och alltid funnits till hands.

Hela gruppen har hjälpts åt med de olika momenten men Viktoria Granli har haft ansvar för rapportskrivningen medan Caroline Abrahamsson och Åsa Sandberg ansvarat för kodningen. Rapport och kodning har skett parallellt.

I rapportens kapitel 1 och 2 har vi utgått ifrån vår planeringsrapport som vi gjorde i kursen System- och Vetenskapsteori.

Arbetet med vårt ex-jobb har varit mycket roligt och givande, vi känner att vi utvecklats både som grupp och individer.

Tack för oss!

Caroline Abrahamsson, Viktoria Granli & Åsa Sandberg

Trollhättan 2002

Innehållsförteckning

1 Inledning	1
1.1 Bakgrund	1
1.2 Problemformulering	1
1.3 Förutsättningar	1
1.4 Syfte	2
1.5 Avgränsningar	2
2 Vetenskapligt synsätt och metod	2
2.1 Metodansats	2
2.2 Forskningsmetodik	3
2.3 Metoder	3
2.3.1 Angreppssätt	3
2.3.2 Metodval	4
2.4 Utvecklingsmiljö	4
2.5 Genomförande	5
2.6 Litteraturstudie	6
3 Teoretisk referensram	6
3.1 FA/SIM	6
3.2 Databasdesign – SUDDmetoden	7
3.2.1 Objektmodellering	7
3.2.2 Normalisering	7
3.2.3 Datamodell med volymeräkning	8
3.3 phpMyAdmin	8
3.4 PHP	8
3.5 MySQL	9
3.6 Klasser och objekt	9
3.6.1 Klasser	9
3.6.2 Arv (extends)	9
3.7 Ramverk	10
3.8 Prototyping	10
3.8.1 Evolutionär prototyping	11
3.8.2 ”Kasta bort” prototyping – throw-away prototyping	11
3.8.3 Test av prototyp	11
3.9 Användargränssnitt	11
4 Resultat	12
4.1 Sitemap	12
4.1.1 gaia_login.inc	12
4.1.2 index.php	12
4.1.3 editfile.php	13
4.1.4 edituser.php	13
4.1.5 editmsg.php	13
4.1.6 adduser.php	14
4.1.7 showmsg.php	14

4.1.8	addmsg.php	15
4.1.9	archive.php	14
4.1.10	uploadfile.php	15
4.1.11	Hjälpfunktioner	15
5	Slutsatser.....	16
5.1	<i>Analys av resultat.....</i>	<i>16</i>
5.2	<i>Rekommendationer till fortsatt arbete.....</i>	<i>17</i>
6	Referensförteckning	17
	Appendix 1	
	Appendix 2	
	Appendix 3	
	Appendix 4	
	Appendix 5	
	Appendix 6	
	Appendix 7	
	Appendix 8	
	Appendix 9	

1 Inledning

Skaraborgs Städ AB skriver på sin hemsida (URL 1) att de är det största svenska privatägda städföretaget i Västra Götaland. Företaget erbjuder ett antal tjänster inom entreprenadstäd, fönsterputs samt specialservice.

Enligt Skaraborgs Städs hemsida (URL 1) har de idag uppdrag hos 250 kunder i Västra Götaland. De har också ökat sin omsättning i snitt med 20 % per år de senaste åren och beräknar omsätta 60 miljoner i år (2002).

Företaget består av totalt fyra kontor. Huvudkontoret är placerat i Moholm och de övriga tre i Falköping, Mariestad och Skövde.

1.1 Bakgrund

Vår uppdragsgivare, Skaraborgs Städ AB vill utveckla och förnya sin webbplats samt göra den mer funktionell än den befintliga. Eftersom inte alla kontor har tillgång till Internet så har det tidigare inte varit aktuellt att utveckla en dynamisk webbsida. Det är först nu som detta blir önskvärt eftersom internetuppkoppling till samtliga kontor beräknas till sommaren 2002. Utvecklingen av webbsidan strävar i första hand mot att förenkla kommunikationen mellan kontoren och riktar sig i dagsläget inte till allmänheten. Uppdragsgivarna har svårt att precisera exakt vad de vill ha ut av sin nya webbplats. Detta mest på grund av att de inte riktigt vet vad en webbplats kan erbjuda för möjligheter.

1.2 Problemformulering

Uppdragsgivaren har gett oss uppgiften att undersöka hur man kan underlätta kommunikationen mellan personalen och mellan de olika kontoren, samt minimera dubbelarbetet. Vi kommer också att realisera hela eller delar av deras behov genom att skapa en färdig produkt.

1.3 Förutsättningar

Företaget har redan en statisk hemsida. Hemsidan har en ändamålsenlig utformning som vår uppdragsgivare är nöjd med. Nu vill de ha en mer dynamisk sida som i första hand ska vända sig till personalen på företaget. Vi fick använda ramarna, bakgrunden och bilderna som redan fanns, för att få ett enhetligt utseende. Vi har dock ändrat designen för att anpassa den till vår applikation och dess syften.

1.4 Syfte

Vårt arbete på företaget kan delas in i två delar. Det primära syftet är att, genom en noggrann analys, undersöka hur man på Skaraborgs Ståd AB, kan underlätta kommunikationen mellan personal och kontor. Det sekundära syftet är att skapa en webbapplikation grundat på vår analys.

Syftet med rapporten är att visa vilka metoder vi använder och hur vi genomför undersökningarna, den ska även visa hur vi skapar en lösning på det problem vi analyserat fram.

1.5 Avgränsningar

Vår uppdragsgivare har ett antal önskemål och idéer om hur de kan utveckla sig, ett är t ex att lägga upp lagerinformation i en databas. Vi har avgränsat oss till att enbart ta hand om deras kommunikationshantering, (eftersom vår problemställning handlar om just kommunikationen mellan kontoren). Några av personalens önskemål kommer att uppfyllas genom det nätverk och den internetuppkoppling som är planerad, och känns därför överflödiga för oss att ta tag i.

Vi kommer inte att erbjuda support eller vidareutveckling av webbapplikationen efter det att vårt arbete är slutfört. För att underlätta för administrationen av applikationen kommer en manual för administratörerna (Appendix 1) att ingå i dokumentationen. Vi ansvarar inte heller för backup av information (databasen) eller buggar som upptäcks i efterhand.

2 Vetenskapligt synsätt och metod

2.1 Metodansats

Vi använde oss av en induktiv metodansats där insamlingen av informationsmaterialet skedde genom muntliga intervjuer/diskussioner med utvalda personer hos uppdragsgivaren. Det insamlade materialet analyserades och dokumenterades och på så sätt fick vi fram fakta om uppdragsgivarens kommunikationshantering och dubbelarbete. Denna ansats används oftast tillsammans med den kvalitativa metoden (Wallén, 2000).

Vi valde bort deduktiv metodansats på grund av att den kräver stor förkunskap om problemet man ska undersöka samt att vi var tvungna att utgå ifrån verkligheten i vår undersökning. Eftersom resultaten från analysen inte kommer att vara mätbara ansåg vi att denna ansats inte passade vår problemställning.

2.2 Forskningsmetodik

Eftersom vårt primära problem handlade om att undersöka hur man skulle kunna underlätta kommunikationen mellan personalen på företaget och de olika kontoren ansåg vi att en explorativ studie passade bäst för att lösa problemet. Vi genomförde studien genom att besöka företaget för intervjuer. Det insamlade materialet skrevs ned och sedan utreddes det med hjälp av problemanalys.

En explorativ studie gör man för att få grundläggande kunskap om problemet, när problemet är ett problem, varför det är ett problem och i vilket sammanhang problemet uppstår. Forskaren ska också utreda vad som inte hör till problemet och vilka alternativ som kan uteslutas (Wallén, 1996).

2.3 Metoder

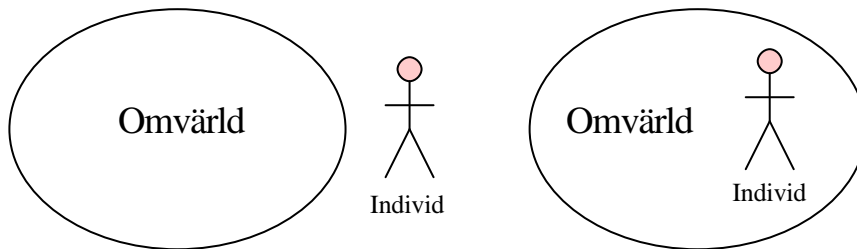
Det är mycket viktigt att välja ut rätt metoder för att kunna utföra en saklig undersökning eller ett forskningsarbete, anser Holme & Solvang (1991). Metoden är också grunden för ett organiserat och metodiskt arbete, den kan ses som ett redskap för att lösa problem och komma fram till ny kunskap (Holme & Solvang, 1991).

2.3.1 Angreppssätt

Den kvalitativa metoden är mindre formaliserande, den beskriver helheten av sammanhanget och har en närhet till källan. Forskarens uppfattning och tolkning är viktig och koncentrationen inriktas på några få enheter (Holme & Solvang, 1991). Kvalitativa studier är nödvändiga för sådant som är vagt och mångtydigt och används i syfte för att uppnå andra värden än medicinska eller tekniska. Det är inte bara något för en avgränsad del av samhällsvetenskap, utan något som berör alla de forskningsområden som lutar åt praktisk verksamhet (Wallén, 1996). Personliga och situationsberoende omständigheter har stor inverkan, och för att ta reda på människors upplevelser får man helt enkelt fråga dem. Wallén (1996) menar att en enkät inte fungerar så bra i dessa situationer. Istället kan en personlig intervju med tematiska ramar tillämpas, där man kan ställa följdfrågor och som forskare bör man bara leda den intervjuade.

Den som intervjuar ska inte agera expert utan person. Det är viktigt att det man samtalar om är relevant. Det som berör viktiga problem ska ske i faktiska situationer och projektet måste vara meningsfullt för dem som berörs (Wallén, 1996).

Holme & Solvang (1991) tar även upp det kvantitativa angreppssättet, som är mer strukturerat, alltså mer planering och kontroll från forskarens sida. Information omvandlas till statistiska analyser och mätbara resultat. Andra kvantitativa kännetecken är att man eftersträvar maximalt god avspegling. Man har en jag-det-relation mellan forskaren och den undersökta. Man skaffar information på ett sätt som präglas av avstånd och urval, det är viktigt att informationen är pålitlig (Holme & Solvang, 1991).



Figur 2.3.1.1 Det kvantitativa och kvalitativa perspektivet (Backman, 1998)

2.3.2 Metodval

Vi ansåg att den kvalitativa metoden passade vårt uppdrag bäst eftersom kvalitativ metod kännetecknas av att den inte är lika formaliserande. Vi genomförde kvalitativa intervjuer där vi försökte skapa diskussioner. Eftersom vi skapade diskussioner, med endast ett fåtal ur personalen, passade inte den kvantitativa metoden där man använder sig av många undersökningsenheter och förutbestämda frågor, t ex i enkätform. Vi ansåg att den typen av undersökning blev för opersonlig och inte skulle leda fram till rätt lösning på uppdragsgivarens problem.

2.4 Utvecklingsmiljö

Prototypen och den färdiga applikationen kördes i Linux-miljö på qube3 som finns på HTU. Vi arbetade med programspråket PHP, databashanteraren MySQL och med JavaScript och HTML.

Vi valde PHP och MySQL av flera orsaker, det första var att stöd för dessa båda fanns på webbhotellet där uppdragsgivarens nuvarande hemsida ligger. Det andra var att både PHP och MySQL är så kallad open source och alltså gratis. JavaScript har vi använt för att hantera kontrollfönster och HTML för designen och strukturen på systemet.

För att skriva koden använde vi ConText och Vi-editorn, filer och bilder laddades upp med Secure Shell till qube3, en server som skolan tillhandahåller. Rapporten skrevs i Microsoft Word 2000.

Vi har använt applikationen phpMyAdmin för att underlätta hanteringen och dokumentationen av databasen (URL 2).

2.5 Genomförande

För att underlätta arbetet och ha tillgång till de senaste upplagorna av vårt material ordnade vi ett konto på Disco (ett gratis interaktivt internet-verktyg), där vi laddade upp filerna när dagen var slut. Rapportskrivandet har skett parallellt med alla de olika momenten.

Första veckan ägnades åt att förbereda frågor och annat inför det första besöket hos våra uppdragsgivare. Väl ute på plats hos uppdragsgivarna upptäckte vi att en del frågor som ingick i vår intervjumall (Appendix 2) var överflödiga och tog därför bort dessa. Innan intervjuen med Fredrik, som är chef för Skaraborgsstad AB, var vi inne på deras hemsida och tog reda på vilka olika formulär som låg ute och tittade över hur man skulle kunna förbättra och förnya sidan. Något som vi lade märke till var att t ex intresseanmälan skickades med e-post. Det första vi tog upp i intervjuen var att fråga hur de behandlades när de kom till företaget.

Informationen vi fick genom intervjuerna gav ett bra underlag att jobba med och respondenterna var entusiastiska och kom med många bra förslag (Appendix 3). Vi strävade efter att inte ställa ledande frågor, utan få respondenterna att tänka efter och komma fram till vad de saknade. Efter besöket sammanställde vi frågorna skriftligt. Vi gjorde en problemanalys (Appendix 4) och vi fick fram det primära behovet, kommunikationen behövde förbättras mellan de olika kontoren. Vi beslutade att göra ett slags meddelandesystem och filarkiv.

Eftersom företaget redan hade en befintlig hemsida, använde vi en del av den existerande designen men gjorde ändringar för att kunna utnyttja dynamiken i PHP. Bilden överst på sidan har vi tagit själva, den föreställer kontoret i Moholm. Bilden på städmoppen har vi fått lov att låna (URL 3). Vi gjorde en prototyp som respondenterna kunde använda och ge synpunkter utifrån innan vi började skapa systemet på allvar. När prototypen var klar för testning skickade vi e-post till respondenterna för att få synpunkter från dem. Vi bifogade även en fil där vi skrivit frågor om varje enskild sida, där de kunde skriva ner sina åsikter och allmänna kommentarer (Appendix 5). Efter tre dagar fick vi tillbaka åsikterna på prototypen från uppdragsgivaren (Appendix 6). Vi gick igenom dessa åsikter och genomförde en del ändringar innan vi besökte våra uppdragsgivare för att diskutera igenom hur den färdiga applikationen skulle se ut. Vid besöket hos uppdragsgivaren gick vi igenom prototypen noga tillsammans med de ur personalen som vi tidigare intervjuat.

Planeringen av systemet var nu klar och det var "bara" att börja koda. För att inte koden skulle bli för rörig kastade vi prototypen. Vi kände att det var lättare att strukturera upp koden på det sätt om vi började om från början, naturligtvis återanvände vi vissa delar. Vi funderade på om vår applikation skulle ha något namn och döpte den efter en stunds funderingar till T.I.S (Trasan Information System).

De sista veckorna har främst ägnats åt kodning. Vi tog kontakt med Magnus Jonsson på företaget Excense. Han gav oss förslaget att använda ett ramverk för att underlätta kodningen av sidorna. Ramverket GAIA (URL 8) som vi använde oss av, gav Magnus Jonsson oss tillgång till och han hjälpte oss även att använda det på ett effektivt sätt. Eftersom vissa funktioner som skulle användas på flera sidor var snarlika använde vi oss av objektorienterad kodning. När vi började arbeta med sessioner och autenciering fick vi byta ut namnen på en del sidor. Om man försöker få upp en av de skyddade sidorna ska inloggningssidan komma automatiskt och skicka användaren till start-sidan efter inloggning, alltså har "start.php" fått byta namn till "index.php" och "index.php" har bytt namn till "gaia_login.inc".

2.6 Litteraturstudie

För att säkra kvaliteten på vårt arbete och för att öka användarvänligheten på den slutgiltiga produkten kommer vi att studera litteratur om

- databasdesign
- programspråk för webbapplikationer
- om användning av "prototyping"
- FA/SIM

Dessa studier kommer att leda oss fram till vilken lösning som både tekniskt och användarmässigt blir den bästa för uppdragsgivarna.

3 Teoretisk referensram

3.1 FA/SIM

För att få fram vilket som är det största behovet hos Skaraborgs Städ valde att analysera den information vi fått under intervjuerna enligt FA/SIM-modellen (Appendix 4). Modellen består bl a av en problemlista som innehåller formuleringar av problem och problemfaktorer.

Problemen analyseras i en problemgraf som beskriver en samlad problemstruktur (en sammanhängande grupp av problem). Man beskriver orsak-effektrelationer mellan problemen för att kunna se hela den aktuella problembilden. Mållistan identifierar mål som är relevanta för problemområdet, målsambanden dokumenteras i en målgraf, olika delmål ska bidra till uppfyllandet av ett överordnat mål. (Goldkuhl Röstlinger, 2000).

3.2 Databasdesign – SUDDmetoden

SUDD (Strukturerad analys och DatabasDesign) beskrivs som ett metodiskt hjälpmedel när man ska bygga databaser. Metoden beskriver ett systematiskt sätt att strukturera och organisera information till data i register (Jonegård, 1989).

Man kan dela in metoden i fem steg, av vilka vi valde att göra två: objektmodellering och normalisering (Appendix 7).

3.2.1 Objektmodellering

En objektmodell består av objekt (objekttyper), relationer och egenskaper. Man kan säga att objekmodellen är en avbild av företeelser man vill ha med i databasen. De objekttyper man väljer att ta med i en objektmodell är de som man vill lagra information om (Jonegård, 1989).

Mellan objekttyperna finns det relationer som kan vara en-till-en, en-till-många eller många-till-många relationer. Ett objekt som har många-till-många relationer ritas som en fyrkant, övriga som cirklar. Varje objekt har också ett visst antal egenskaper (termer eller attribut). Varje objekttyp måste ha en identifierare (unik egenskap, t ex ett personnummer). Objekttypens identifierare brukar märkas, te x med en stjärna.

Till varje objektmodell ska det finnas en objekt-term-katalog och en termkatalog, som bl a visar vilka egenskaper ett objekt har. Termkatalogen har också som uppgift att ge en god översikt över de termer som återfinns i databasen (Jonegård, 1989).

3.2.2 Normalisering

Normalisering gör man för att kontrollera att varje egenskap enbart lagras på en plats, alltså för att förhindra dubbellagring (redundans) i databasen. Vilka egenskaper som ska lagras i databasen känner man till genom objektmodelleringen som gjorts tidigare.

Varje objektens egenskaper identifieras, varefter man sätter ihop egenskaper med samma identifierare till logiska poster. De logiska posterna kontrolleras mot

de tre normalformerna (av vilka vi använt de två första) för att upptäcka och reducera redundans i databasen. (Jonegård, 1989).

3.2.2.1 1: a normalformen (1:a NF)

En logisk post befinner sig på 1:a normalformen (1:a NF) om dess termer är odelbara och återfinns endast en gång i posten (Jonegård, 1989).

3.2.2.2 2: a normalformen (2:a NF)

För att en logisk post ska befinna sig på andra normalformen (2:a NF) ska 1:a NF vara uppfylld, och alla postens egenskapstermer ska vara helt beroende av hela primärnyckeln (Jonegård, 1989).

3.2.3 Datamodell med volymberäkning

Efter normaliseringen kan en datamodell med volymberäkning göras för att visa hur stor databasen skulle kunna bli (Jonegård, 1989).

3.3 phpMyAdmin

Vi visste att det skulle bli mycket jobb med databasen. Därför ville vi använda oss av ett webbaserat grafiskt gränssnitt att hantera databasen med. Vi valde phpMyAdmin (URL 2) som är gratis, och laddade ned det till qube3 och installerade. Denna applikation är enkel att använda och bra om man vill skriva ut dokumentation om databasen (Appendix 8). För att höja säkerheten satte vi ett lösenordskydd för att förhindra intrång i databasen via phpMyAdmin-applikationen.

3.4 PHP

Viktor Jonsson (URL 4) skriver att i en omröstning på PHPs officiella webbsida ansåg en majoritet på 53 procent att PHP är en förkortning för: PHP Hypertext Preprocessor.

PHP är ett HTML-inbäddat skriptspråk som körs på serversidan och distribueras som öppen källkod (Open Source). Koden skrivs direkt i HTML-dokumentet och interpreteras (tolkas) av webbservern till HTML-kod. Detta sker innan webbsidan skickas över till användaren.

En vanlig tolkning av PHP är enligt Jonsson att det är UNIX motsvarighet till Microsofts ASP. PHP kan köras med bland annat Apache, Roxen, Java (servlet), IIS (även PWS), AOLserver och som CGI (URL 4).

PHP är enligt Viktor Jonsson ett mycket bra programmeringsspråk för att hantera databaser. PHP har stöd för ett stort flertal databaser och det finns

många bra funktioner som gör hanteringen enkel. PHP har stöd för bland annat databashanterarna Oracle, Sybase, PostgreSQL och MySQL (URL 4).

3.5 MySQL

MySQL är en DBMS (Database Management System) alltså ett program som manövrerar databaser. Ibland kallas program av den här typen för databasservrar eller enbart databaser, men det är enligt Viktor Jonsson (URL 5) viktigt att skilja på databas och DBMS. En databas är en samling data som är strukturerade på ett speciellt sätt, medan en DBMS är ett program som hanterar databasen. Viktor Jonsson anser att MySQL är den dominerande databasservern inom öppen källkods rörelse. MySQL går bra att använda i ett flertal operativsystem. MySQL är grundat på klient/server-tekniken, det betyder att systemet är uppdelat i server och klient. Servern är huvuddelen av systemet och klienten är den del som användaren behöver för att kommunicera med servern. Jonsson skriver att en stor fördel med klient/server-metoden är att det går att installera delarna på olika datorer. Det är heller inte nödvändigt att servern och klienten körs på samma operativsystem. MySQL kan i princip användas tillsammans med vilka programmeringsspråk som helst. Det finns dock en del programmeringsspråk, exempelvis PHP, som har inbyggt stöd för MySQL (URL 5).

3.6 Klasser och objekt

3.6.1 Klasser

När man vill samla olika funktioner på en enda plats är det mycket användbart att använda sig av klasser. Ifall man t ex har en mängd funktioner för databashantering, kan det vara smidigt att skapa en klass för att samla funktionerna för att få en bättre struktur på koden.

Funktionerna (metoderna) i klassen är de som skickar, hämtar och behandlar data. De är själva kärnan för att en klass ska kunna fungera som en riktig klass. Funktionerna (metoderna) skrivs som vanliga funktioner, och anropas via instansen av klassen (URL 6).

3.6.2 Arv (extends)

Man behöver ofta klasser med liknande funktioner och variabler, som redan finns i existerande klasser. Då kan det vara bra att definiera en generell klass som man sedan kan använda i senare projekt. För att åstadkomma detta kan klasser vara utökningar av andra klasser. Den utökade klassen har alla variabler och funktioner som bas klassen. Detta kallas för arv. Det är inte

möjligt att ärva bara vissa funktioner från en klass utan en klass ärver alltid hela basklassen (URL 7).

Genom att använda oss av extends, dvs arv i våra klasser fick vi även tillgång till de funktioner som finns i klasserna i ramverket (Appendix 9, t ex klassfilen tis_category.class).

3.6.3 Ramverk

GAIA är ett lättvikts PHP-applikationsramverk med några delar från PHPLIB. Magnus Jonsson (URL 8) skriver att syftet med ramverket GAIA är att erbjuda en funktionell kodbas som kan användas för att skapa webbsidor och webbapplikationer. Ramverket består av ett antal klasser och en konfigurationsfil. I konfigurationsfilen finns ett antal konstanter definierade som vi ändrade efter våra behov. För att slippa ändra sökvägar i alla filer i efterhand, om filerna ska flyttas till en annan server, så har vi definierat konstanter som innehåller den definitiva sökvägen. Detta medför att man bara behöver ändra på endast ett ställe, och det är i gaia.conf (Appendix 9).

Exempel på hur det kan se ut:

include(_GAIA_WEBROOT_DIR.'header.php'), där _GAIA_WEBROOT_DIR är konstanten och header.php är filen. Andra saker som vi tagit GAIA till hjälp med är databaskoppling och inloggning samt autenciering.

3.7 Prototyping

Det är svårt för användare att vid de första intervjuerna specificera vad han/hon vill att programvaran skall göra och vad det får för konsekvenser. Prototyper skapas för att kund och utvecklare på ett bättre sätt skall förstå kraven. För att prototyper skall vara effektiva måste de utvecklas snabbt så att den blir billig och användarna kan testa den tidigt i utvecklingsprocessen. Några av prototypens främsta fördelar är: förbättrad användbarhet för färdig produkt, bättre anpassning efter kraven, förbättrad designkvalitet, förbättrat underhåll, minskad ansträngning vid utveckling (Sommerville, 2001).

Fördelar	Nackdelar
Snabbare leverans	Problem men ledning- ledningen är inte utbildad på denna typ av utveckling
Användare är mer engagerade i programvaran.	Problem med underhåll- snabb utveckling tenderar till att korrumpera strukturen.

Figur 3.7.1 Fördelar & nackdelar med prototyping (Sommerville 2001)

3.7.1 Evolutionär prototyping

Man utvecklar en prototyp genom iterationer. En första prototyp visas för användaren, samma prototyp förfinas och blir bättre. Detta angreppssätt används främst när man inte på ett bra sätt kan precisera vilka krav det skall finnas på en programvara. Verifiering är omöjlig eftersom det inte finns någon specifikation, validering sker genom demonstrering av prototypen för användarna (Sommerville, 2001).

3.7.2 ”Kasta bort” prototyping – throw-away prototyping

Sommerville (2001) visar på att den huvudsakliga funktionen för denna typ av prototyping är att klargöra kraven och samt riskerna för ledningen. Efter evaluering så kasseras hela prototypen. Detta tillvägagångssätt används ofta när hårdvarusystem utvecklas. Man skall inte använda denna prototyp och bygga vidare på den eftersom det finns risk att väsentliga delar fattas. Det är inte heller säkert att prototypen kommer att ha samma funktionalitet som det färdiga systemet. Man kan i vissa fall återanvända vissa komponenter från prototypen.

3.7.3 Test av prototyp

Eftersom de användare som vi utvecklar webbapplikationen för, har liten eller ingen datavana, valde vi att lägga lite extra tid på att skapa en prototyp som i deras ögon är ganska lik den slutgiltiga produkten. Genom att visa upp en prototyp med några få fungerade funktioner och inte bara gränssnitt trodde vi att det skulle bli lättare för dem att beskriva för oss vilka funktioner och vilket utseende de ville ha. För att de inte skulle känna sig stressade av att vi stod med när de testade webbapplikationen valde vi att skriva ihop ett dokument som de enkelt kunde skriva ut och använda som en utvärderingsmall av prototypen. Utvärderingen gjordes av fem personer på företaget och var de som vi redan haft kontakt med.

3.8 Användargränssnitt

För att göra systemet användarvänligt och öka lusten att använda det, har vi lagt in hjälp-funktioner på varje sida som alla vanliga användare har tillgång till. Administratörerna till systemet får en användarmanual för specifika sidor som enbart dessa har tillgång till. Systemets design liknar till viss del uppdragsgivarnas officiella hemsida, t ex färgsättningen.

4 Resultat

4.1 Sitemap

- Gaia_login.inc
 - index.php
 - editfile.php
 - edituser.php
 - editmsg.php
 - showmsg.php
 - addmsg.php
 - archive.php
 - uploadfile.php

Figur 4.1.1 Översikt av T.I.S – Trasan Information System. Koden finns i (Appendix 9).

4.1.1 gaia_login.inc

En sida med mycket enkel design. Den består av två textfält och en "Logga in"-knapp. För att kunna logga in måste användaren registrera sig och detta görs av en utsedd administratör. Vid registrering får användaren ett användarnamn och ett lösenord.

4.1.2 index.php

Länkar



Administratörlänkar

- [Lägg till användare](#) - Registrera en ny användare
- [Editera/Ta bort användare](#) - Ta bort eller ändra uppgifter
- [Ta bort meddelanden](#) - Ta bort meddelanden
- [Ta bort filer](#) - Ta bort inaktuella dokument

Länkar

- [Meddelandesystemet](#) - Skriv och läs meddelanden
- [-Skriv nytt meddelande](#) - Gör ett inlägg i meddelandesystemet
- [Filarkiv](#) - Protokoll, nyhetsbrev och mallar
- [-Ladda upp fil](#) - Skicka upp en fil, t ex protokoll, nyhetsbrev...

Denna sida är hjärtat i systemet och hit skickas man automatiskt när man loggat in. Länkarna på sidan anpassas beroende på om man är inloggad som administratör eller vanlig användare. Vid inloggning som administratör visas länkar till sidor som har med registrering, editering och radering av poster att göra och

Figur 4.1.2.1 Skärmdump från T.I.S (index.php)

även de länkar som alla användare får tillgång till. Anledningen till att de vanliga användarna inte får så många rättigheter som t ex att radera en fil, är att de är relativt ovana vid att använda sig av datorer och på så sätt försöker vi minimera riskerna med oavsiktlig radering och editering.

4.1.3 edituser.php

En sida där administratörer kan gå in och ändra uppgifter om användaren. Överst på sidan finns en rullgardinsmeny där alla användarposter visas med för- och efternamn läses in från databasen. Vi har valt att sortera alla poster i bokstavsordning baserat på användarnas efternamn. När man hittat den användare som man ska ändra någon personuppgift på visas användaruppgifterna i textfälten. I fälten kan man fylla i de nya uppgifterna och sedan trycka på knappen "Uppdatera". Här ligger posten kvar i fälten för att administratören ska kunna se att posten verkligen uppdaterats. Vill man radera en användare trycker man på knappen "Ta bort". Här har vi en kontroll som frågar om man verkligen vill ta bort posten.

Ta bort/editera användare

Användare	<input type="text" value="Sandberg Åsa"/>
Förnamn*	<input type="text" value="Åsa"/>
Efternamn*	<input type="text" value="Sandberg"/>
Telefonnummer	<input type="text" value="0521-65506"/>
Epost	<input type="text" value="asaevsandberg@hotmail.com"/>
Användarnamn*	<input type="text" value="assan"/>
Lösenord*	<input type="text" value="kkyky"/>
Befattning*	<input type="text" value="Resurs"/>
Kontor*	<input type="text" value="Skövde"/>

Figur 4.1.3.1 Skärmdump från T.I.S (edituser.php)

4.1.4 editmsg.php

På den här sidan kan administratören gå in och ta bort befintliga meddelanden. Inläggen visas i kronologisk ordning, d v s de senast skrivna visas överst. Man tar bort ett meddelande genom att trycka på "Ta bort"-knappen som hör till det meddelandet man vill radera. Även här kommer en kontroll-ruta upp för att bekräfta raderingen.

4.1.5 editfile.php

Detta är en sida för endast administratörer. Här kan man ta bort filer som är inlagda i databasen (filer som blivit uppladdade). För att enkelt hitta den fil

man önskar radera, har vi valt filkategori som en begränsande faktor. Kategorierna är baserade på företagets användning av olika dokument. De olika kategorierna är: Mötesprotokoll, Mallar, Scheman, Nyhetsbrev och Övrigt. De poster som stämmer överens med begränsningen som görs, visas efter varandra, där varje post har en "Ta bort"-knapp. Denna knapp har en kontroll som medför att man måste svara ja för att filen verkligen ska raderas.

4.1.6 adduser.php

Ytterligare en sida för administratörerna. För att en användare ska kunna använda systemet måste de vara registrerade. Registreringen görs av administratören på begäran av användaren. De fält som är markerade med * är obligatoriska fält och måste fyllas i för att registreringen ska genomföras. Om användaren har en e-postadress skickas ett autogenererat brev till den registrerade med dess uppgifter som bl a användarnamn och lösenord. Har användaren ingen e-post måste administratören personligen meddela användaren om de uppgifterna som blivit lagrade. För att undvika alltför enkla lösenord finns det även här en kontroll, lösenordet måste vara minst fem tecken långt för att accepteras.

4.1.7 showmsg.php

Meddelandesystemet är en sida där användarna kommer att kunna se meddelanden skrivna av sina kollegor. Inläggen visas med en tydlig rubrik, datum, avsändare och avsändarens befattning. Längst upp på sidan finns en länk till sidan addmsg.php (skriva nytt inlägg) så att man enkelt kan skriva ett eget meddelande. För att

Meddelandesystemet



[Skriv nytt inlägg](#)

Hejsan!

2002-05-23 19:01

Hur går detta??

Ewa Abrahamsson
Administratör

Figur 4.1.7.1 Skärmdump från T.I.S (showmsg.php)

underlätta för administratören har vi gjort en funktion som automatiskt väljer att inte visa meddelanden som är äldre än två månader.

4.1.8 archive.php

En sida där man enkelt kan se de filer som är uppladdade. För att lättare hitta just den filen som man letar efter kan man välja en bestämd fil-

kategori. Dess kategorier är samma som nämndes i kap 4.1.5. Alla tillgängliga filer, enligt vald kategori, visas i en lista där de visas med beskrivning, filnamn och uppladdningsdatum.

4.1.9 uploadfile.php

För att kunna skicka upp en fil gjorde vi en sida för uppladdning. Sidan består av ett antal fält som alla är obligatoriska att fylla i. Det är viktigt att varje användare skriver en beskrivning till filen de skickar upp. Filen ska placeras i en kategori som man väljer själv i en rullgardinsmeny, t ex. Scheman. Man måste också fylla i vilka olika befattningar som ska få åtkomst till filen. Dessa väljer man genom att kryssa i checkboxarna. Genom att trycka på "Skicka"-knappen placeras den aktuella filen i databasen.

Filarkivet



[Ladda upp ny fil](#)

Filkategori

Beskrivning	Alla ska få se den här!
Filnamn	rapport.doc
Fildatum	2002-05-23 14:38:00

Figur 4.1.8.1 Skärmdump från T.I.S (archive.php)

4.1.10 addmsg.php

Denna sida är enkel men tydlig. Vill man skriva ett eget meddelande fyller man i rubriken som är obligatorisk och sedan skriver man sitt meddelande. Avsändaren till meddelandet blir automatiskt den som man är inloggad som. På "Skicka"-knappen finns en funktion som kontrollerar så att man verkligen fyllt i obligatoriska fält. Det finns även en "Rensa"-knapp som tömmer fälten om användaren bestämmer sig för att inte skicka meddelandet.

4.1.11 Hjälpfunktioner

Alla sidor som vanliga användare har tillgång till har hjälpknappar som användaren kan klicka på om de behöver hjälp, dessa sidor är: "addmsg.php", "showmsg.php", "uploadfile.php", "archive.php", "index.php"

och "gaia_login.inc". Sidorna som administratörerna har tillgång till har inga hjälpknappar, till dessa sidor finns en användarmanual.

5 Slutsatser

5.1 Analys av resultat

Genom att använda oss av ett redan testat och stabilt ramverk hoppas vi på att vi skapat mer stabil och säker applikation, även om vi är mycklet medvetna om att den aldrig kommer att komma i närheten av att bli buggfri. När det gäller inloggning så är det mycket viktigt att det finns en god säkerhet inbyggd, speciellt när viktig företagsinformation lagras i en databas.

Det färdiga resultatet blev en dynamisk webbapplikation efter personalens behov, och önskemål. Eftersom vår uppdragsgivare är ett städföretag tyckte vi att T.I.S - Trasan Information System var ett passande namn. I systemet ingår ett filarkiv, där användarna kan ladda upp filer som t ex mötesprotokoll, nyhetsbrev, scheman, bilder etc. Detta blir ett miljövänligt och billigt alternativ till att skicka t ex nyhetsbrevet en gång i månaden till alla anställda i pappersformat vilket sker i dagsläget.

Om t ex en arbetsledare är sjuk eller en mötestid ändrats rings det runt i hela företaget, vi har gjort ett meddelandesystem som en alternativ lösning på det problemet. I meddelandesystemet kan alla användare göra inlägg och läsa andra användares inlägg, om samtliga i företaget tar för vana att gå in och läsa inläggen kan kommunikationen förenklas avsevärt och en mängd dubbelarbete undvikas. Meddelandesystemet visar enbart de inlägg som är gjorda de senaste två månaderna för att hålla systemet aktuellt. Administratören kan däremot se alla inlägg som gjorts på den sidan där inläggen raderas.

För att underlätta editeringen av innehållet i databasen har vi gjort ett antal sidor som endast administratören har tillgång till. Sidorna är till för att lägga till och editera användare, samt ta bort meddelanden och filer. Vi har gjort en enkel användarmanual ifall det är något som är oklart eller om administratören är sjuk och någon annan ska sköta systemet. De sidor som alla användare har tillgång till innefattar hjälpknappar ifall användaren inte förstår hur man ska gå tillväga, t ex vid filuppladdning.

Givetvis kan det bli fel när administratörer lägger till, tar bort eller editerar uppgifter. Sidorna behöver därför viss validering av fält och knappar. För att inte göra det för besvärligt att fylla i formulären har vi valt att validera

enbart det nödvändigaste funktionerna, och dessa är i första hand inmatningsfält och raderingsknappar.

Med applikationen vi utvecklat kan personalen undvika onödiga telefonsamtal, stora utskrifter och massutskick av protokoll och nyhetsbrev. Med detta hoppas vi att uppdragsgivarna kan minska dubbelarbete och förenkla sin kommunikation.

5.2 Rekommendationer till fortsatt arbete

En webbapplikation kan byggas ut hur mycket som helst. En del funktioner som T.I.S skulle kunna byggas ut med är arkivering av gamla inlägg och filer där administratören kan välja mellan att arkivera eller radera inlagd information. Meddelandesystemet skulle också kunna byggas ut så att det finns möjligheter att svara på ett specifikt inlägg.

En annan idé är att använda någon form av kalender där man kan lägga in allt som händer månad för månad på ett lättöverskådligt sätt.

T.I.S skulle också kunna innefatta lagerhantering eftersom applikationen är kopplad till en databas.

6 Referensförteckning

Litteratur

Backman, J. (1998). *Rapporter & uppsatser*. Lund: Studentlitteratur

Eriksson, L T., & Wiederheims-Paul, F. (1999) *Att utreda, forska och rapportera*. Malmö: Liber AB

Goldkuhl, G., & Röstlinger, A. (2000). *Förändringsanalys Arbetsmetodik och förhållnings-sätt för goda förändringsbeslut*. Lund: Studentlitteratur

Holme, I, M., & Solvang, B, K. (1991). *Forskningsmetodik*. Lund: Studentlitteratur

Jonegård, M. (1989). *SUDD metoden strukturerad analys och databasdesign*. Stockholm: Liber

Sommerville, I. (2001). *Software Engineering*. USA: Addison-Wesley

Wallén, G. (1996). *Vetenskapsteori och forskningsmetodik*. Lund: Studentlitteratur

Internetadresser

URL 1:

<http://www.skaraborgstad.se/ftg-omoss.htm>

URL 2:

<http://www.phpmyadmin.net/>

URL 3:

<http://ipp.xorshop.com/ipp/art-113-1.asp>

URL 4:

http://www.redhat.nu/php_introduktion.php

URL 5:

http://www.redhat.nu/mysql_introduktion.php

URL 6:

<http://www.phpportalen.net/moduler.php?modul=skola&kap=9>

URL 7:

<http://php.linux.se/phpdoc/html/keyword.extends.html>

URL 8

<http://chainsaw.phrenetic.net/software/gaia/>

Appendix 1

Användarmanual

MANUAL FÖR ADMINISTRATÖR

T.I.S – TRASAN INFORMATION SYSTEM



INNEHÅLLSFÖRTECKNING

LÄGG TILL ANVÄNDARE	1
TA BORT / EDITERA ANVÄNDARE	3
TA BORT MEDDELANDE	5
TA BORT FILER	7

LÄGG TILL ANVÄNDARE



Lägg till användare

Fält markerade med * är obligatoriska och måste fyllas i

Förnamn *	<input type="text" value="Anna"/>
Efternamn *	<input type="text" value="Petersson"/>
Telefonnummer	<input type="text" value="031-884588"/>
Epost	<input type="text" value="annap@p.se"/>
Användarnamn *	<input type="text" value="annap"/>
Lösenord *	<input type="text" value="panna"/>
Befattning *	<input type="text" value="Städare"/>
Kontor *	<input type="text" value="Falköping"/>

Registrera

Rensa



LÄGG TILL ANVÄNDARE

De fält med en stjärna (*) bakom är obligatoriska och måste alltid fyllas i, annars registreras inte användaren och det dyker upp ett felmeddelande.

- 1. Fyll i förnamn ***
- 2. Fyll i efternamn ***
- 3. Fyll i telefonnummer**
- 4. Fyll i epost**
 - a. Om detta fält fylls i skickas ett autogenererat brev med information om *användarnamn* och *lösenord*
 - b. Om fältet lämnas tomt måste man manuellt informera användaren om dess *användarnamn* och *lösenord*
- 5. Fyll i användarnamn ***
 - a. Användarnamnet ska användas tillsammans med lösenordet vid inloggning
- 6. Fyll i lösenord ***
 - a. Ska bestå av både bokstäver och siffror
 - b. Ska ej vara namn på saker och ting
 - c. Måste vara minst 5 tecken långt
- 7. Välj befattning i rullgardinsmenyn ***
 - a. Viktigt att observera att det endast är ett fåtal användare som ska ha befattningen *administratör* pga säkerheten
- 8. Välj kontor i rullgardinsmenyn ***
- 9. Tryck på knappen "Skicka"**
 - a. Om man vill ångra inmatningar i fälten trycker man på knappen "Rensa"

TA BORT/EDITERA ANVÄNDARE



Ta bort/editera användare

Användare

Förnamn*

Efternamn*

Telefonnummer

Epost

Användarnamn*

Lösenord*

Befattning*

Kontor*

Uppdatera

Ta bort



TA BORT / EDITERA ANVÄNDARE

- 1. Välj den användare som ska tas bort eller editeras från rullgardinsmenyn**
 - a. Användarna är sorterade i fallande bokstavsordning efter efternamn
- 2. Uppdatera fälten genom att först ändra texten och sedan klicka på knappen "Uppdatera"**
 - a. Observera att man inte kan ta bort texten i de obligatoriska fälten, då kommer felmeddelande att visas
 - b. Användarposten kommer att visas med de nya uppgifterna i fälten så att man kan kontrollera att de stämmer
- 3. Ta bort användaren genom att klicka på knappen "Ta bort"**
 - a. För ökad säkerhet och för att undvika misstag måste man svara på en kontrollruta att man verkligen vill ta bort användaren

TA BORT MEDDELANDE



Ta bort meddelande

Sopsortering!

2002-05-22 16:58

Anna Petersson
Städare

Ta bort

Kaffeautomaten sönder

2002-05-22 16:57

anna Qvennerstedt
Administratör

Ta bort

Tappat min kapp!

2002-05-22 16:53

Caroline Abrahamsson
Administratör

Ta bort

Glöm inte att fylla i körjournalen

2002-05-22 16:51

Ewa Abrahamsson
Administratör

Ta bort



TA BORT MEDDELANDE

1. **Här listas automatiskt alla meddelanden som finns och som stämmer överens med den inloggades befattning**
 - a. Är man inloggad som administratör ser man alla meddelanden
2. **För att ta bort ett meddelande trycker man på knappen ”Ta bort” som står i anlutning till det meddelande man vill ta bort**
 - a. För att inte ta bort meddelande av misstag måste man bekräfta raderingen vid en kontrollruta som kommer upp

TA BORT FILER



Ta bort fil

Filkategori

Beskrivning **Möte för planering av sommarschema**
Filnamn 2002-05-22 sommarmöte.txt
Datum 2002-05-22 16:26:00

Ta bort

Beskrivning **Månadsmöte med resurs i April**
Filnamn 2002-04-16.doc
Datum 2002-05-22 16:24:00

Ta bort



TA BORT FILER

- 1. Välj vilken kategori av filer som ska visas genom att klicka på rullgardinsmenyn**
 - a. Filerna som tillhör urvalet listas med beskrivning, filnamn, datum och en "Ta bort"-knapp för varje fil
- 2. Klicka på knappen "Ta bort" vid den filen som ska tas bort**
 - a. När man tar bort ett en fil kommer det upp en kontrollruta för att bekräfta raderingen

Appendix 2

Intervjumanual

Vi presenterar oss

Talar om att vi ska återkomma om ett par veckor

Frågemall:

- Hur många jobbar på de olika kontoren?
- I hur stor utsträckning används datorerna på de olika kontoren?
- På vilket sätt skulle ert arbete vid datorn kunna underlättas t ex hur intresseanmälningarna behandlas, databas, pärm ?
- Intresse av ett forum? Funktioner på det forumet i så fall?
- personlig inloggning
- anslagstavla/meddelandesystem
- intresseanmälan (från hemsidan)
- kontaktformulär (från hemsidan)
- sökbara funktioner
- Hur många kommer i så fall att använda forumet?
- Användarmanual?
- Någon måste ha ansvaret (administrera) över sidan, finns det någon som kan göra detta?
- Förändringar på nuvarande hemsida?
- Prototyp?

Appendix 3

Intervjuer

Fredrik Petrus VD, 2/4 Kontoret i Moholm

På vilket sätt skulle ert arbete vid datorn kunna underlättas t ex hur intresseanmälningarna behandlas, databas och pärmar?

F sa att de togs om hand av Marie och Andrea på administrationen i Moholm. Sedan skickades de till personalchefen i pappersformat. Hon tar kopia på det och skickar originalet till berört kontor. Vi gav förslag på hur man skulle kunna lösa detta genom att de som anmäler sig läggs in automatiskt i en webbaserad databas. Databasen skulle även kunna ha en sökfunktion mm.

I hur stor utsträckning används datorerna på de olika kontoren? Hur många kommer i så fall att använda forumet om det blir ett sådant?

Alla kontoren har inte fått datorer med Internet-uppkoppling än. Det är dock tänkt att alla kontor ska ha tillgång till Internet innan sommaren. Det kommer då att finnas minst två datorer på varje kontor. Ytterligare problem är att personalens datorvana är liten eller ingen alls.

En sak Fredrik tänkt på var hanteringen av ordrar i dagsläget sköts manuellt. En annan sak som är ett problem är att schema ändringar rings eller faxas in och det då blir en massa dubbelarbete.

I framtiden vill han även lägga in lagret i en databas, för bättre översikt på vad som fattas osv.

Intresse av ett forum? Funktioner på det forumet i så fall?

Många anställda kommer aldrig in till kontoret och då vore det bra att kunna lägga ut olika information på nätet. T ex nyhetsbrev.

Företaget är medlemmar i Servicekompagniet, där flera företag i hela landet samarbetar. Där har Fredrik varit inne på diskussionsforum och kollat och tyckte att det verkade bra med något liknande på deras egen hemsida. Han hade funderingar på att det kanske kunde innehålla: Arbetsträffsdatum, protokoll från möten o dyl., mallar, scheman, forum, meddelandesystem osv.

Då skulle det också vara behörighet för olika grupper. Diskussionsforumet är önskvärt för att öka kommunikationen mellan de olika kontoren.

Han pratade även om en anbudssida som skickas till Gustav och han kollar vad det kan kosta och skickar tillbaka kostnadsförslag.

Någon måste ha ansvaret (administrera) över sidan, finns det någon som kan göra detta?

Ja det finns det.

Förändringar på nuvarande hemsida?

Fixa länkarna vore bra.

Prototyp?

Fredrik ville gärna se en prototyp.

Andrea & Marie, administrativ personal, 2/4 – Kontoret i Moholm

Först informerade dem om vad Fredrik sagt under intervjun, vi ville veta vad de hade för åsikter och om det var något de ville lägga till.

Intresse av ett forum? Funktioner på det forumet i så fall?

De ansåg att det vore bra med någon typ av meddelandesystem där man exempelvis kunde skriva in om någon av arbetsledarna var borta för dagen, ändrade mötestider o dyl. Det skulle göra att de slipper ringa en massa extrasamtal.

Hur många kommer i så fall att använda forumet?

Alla har ju inte tillgång till dator och Internet ännu.

På vilket sätt skulle ert arbete vid datorn kunna underlättas t ex hur intresseanmälningarna behandlas, databas, pärm ?

De kommenterade också att om personalens nyhetsbrev ska skickas ut via e-post så måste man ta reda på vilka i personalen som har egna e-postadresser och vilka som kollar sin e-post, övriga behöver få nyhetsbrevet via post som förut.

Prototyp?

Det vore bra

Annelie Eliasson (resurs) & Pernilla Ahl, (städledare) 4/4 – Kontoret i Skövde

Eftersom vi inte hade någon möjlighet att göra en personlig intervju gjorde vi en muntlig intervju via telefon. Det första vi tog upp att diskutera var ifall de visste något sätt att effektivisera deras administrativa arbete.

De nämnde med en gång att de skulle vilja ha ett system för att underlätta att se i slutet av månaden om någon har varit sjukanmäld. De tyckte även att det verkade bra med enkäter av olika slag t ex ledighetsansökningar och ordrar. När de sagt detta frågade vi om det var något de olika kontoren kunde dela information om, på nätet. Vi började diskutera om ett meddelandesystem eller liknande skulle användas av dem och vad som skulle kunna ligga på en sådan sida. De föreslog mötesprotokoll och att det vore bra om de kunde lägga ut om möten ändrades för just nu var det lite så och så med det. Schemamallar och brevmallar ville de ha på sidan. De tyckte

att det vore bra om de kunde gå in på sidan innan de åkte på mötet och kolla en sista gång så att det verkligen var när det var sagt. Just nu kan de inte gå in på nätet och kolla, för de har ingen Internet uppkoppling, men det ska de få innan sommaren. De var mycket positiva till att ta reda på informationen själva, för att de trodde att det skulle bli mindre missförstånd. Nu skrev de Post-it-lappar om ändrade möten, sjukfrånvaro och semester etc.

Appendix 4

FA/SIM analys

Innehåll:	Sid:
Problemlista	2
Problemgraf	3
Mållista	4
Målgraf	5

Problemlista

Utfärdare
CA, VG, ÅS

Datum
2002-04-04

Avser: Informationshantering

Sid 1

- P1** Dubbelarbete
- P2** Schemaändringar o dyl rings eller faxas in till de olika kontoren
- P3** Många telefonsamtal
- P4** Missuppfattningar mellan de olika kontoren
- P5** Onödig pappershantering
- P6** Vet inte om och när folk finns på de olika kontoret
- P7** Många lösa lappar
- P8** Alla har inte tillgång till Internet
- P9** Datavanan är dålig eller obefintlig
- P10** Kommunikationsproblem mellan kontoren
- P11** Nyhetsbrev skickas per post till de anställda
- P12** Dubbellagring av intresseanmälan i pappersformat
- P13** Hantering av ordrar och beställningsjobb sköts manuellt
- P14** Många anställda kommer sällan/aldrig in på kontoret

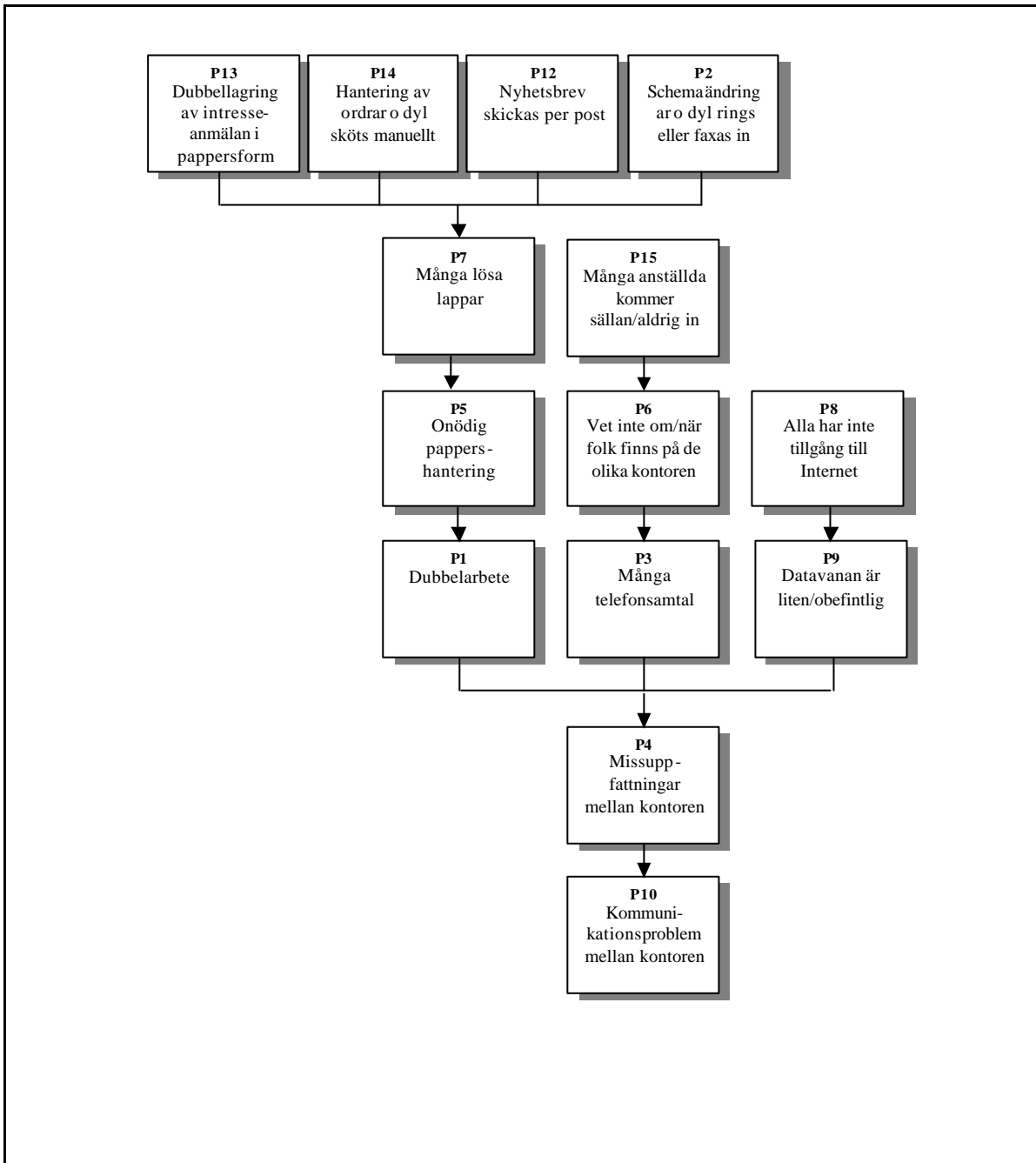
Problemgraf

Utfärdare
CA, VG, ÅS

Datum
2002-04-05

Avser: Informationshantering

Sid 1



Mällista

Utfärdare
CA, VG, ÅS

Datum
2002-04-05

Avser: Informationshantering

Sid 1

- | | |
|------------|---|
| M1 | Anställda hämtar sin information (information-pull) |
| M2 | Datorer och Internet-uppkoppling på alla kontor |
| M3 | Nyhetsbrev på nätet/via e-post |
| M4 | Förenklad/effektivare kommunikation |
| M5 | Säker kommunikation (alla får del av ändringar o dyl) |
| M6 | Enkel manual |
| M7 | Forum för alla anställda |
| M8 | Välantvänt meddelandesystem |
| M9 | Mindre dubbelarbete |
| M10 | Större datavana |

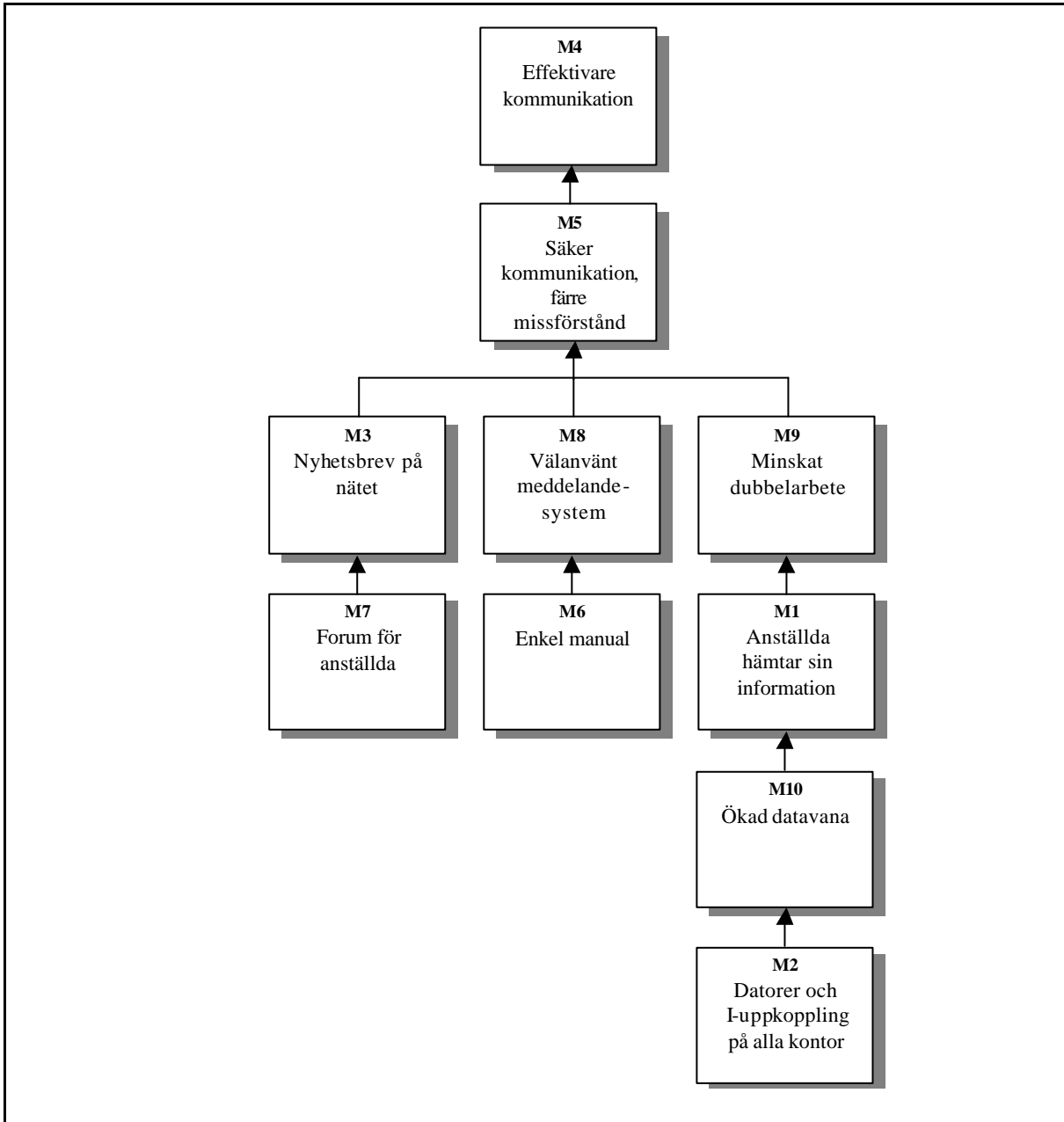
Målgraf

Utfärdare
CA, VG, ÅS

Datum
2002-04-05

Avser: Informationshantering

Sid 1



Appendix 5

Feedback-frågor till prototyp

Namn:

Inloggning (index.php) <http://qube3.thn.htu.se/~examen5/>

På denna sida skriver användaren in sitt användarnamn och lösenord och klickar på "logga in"-knappen och skickas automatiskt till startsidan. Är det en användare utan administratörsrättigheter kommer startsidan att innehålla fyra länkar. Är det en användare med administrationsrättigheter, kommer även administratörlänkarna upp.

Har ni några synpunkter på den här sidan, t ex design, användbarhet, rubriker?

Saknas något, t ex hjälpknapp ?

Övrigt:

Startsida (start.php) <http://qube3.thn.htu.se/~examen5/start.php>

På denna sidan visas länkar beroende på vad man är inloggad som.

Specifika länkar för administratörer:

- Ta bort filer
- Editera användare
- Ta bort meddelanden
- Registrera användare

Länkar för alla, inklusive administratören:

- Meddelandesystemet
- -Skriv nytt meddelande
- Arkiv
- -Ladda upp fil

Administratörlänkar

Saknas det något här? Något mer som endast administratören ska se?

Länkar

Länkar som alla ska se, synpunkter?

Behövs det något mer bland länkarna längst ned i det svarta fältet?

Övrigt:

Ta bort filer (editfil.php)

<http://qube3.thn.htu.se/~examen5/editfil.php>

Här kommer man att kunna välja vilken kategori filen man vill ta bort ska höra till, och när den lades upp. Efter man valt det, trycker man på knappen "hämta fil". Filerna kommer hamna under varandra med en ta bort knapp under varje fil, för att enkelt radera den man vill.

Sökning:

Fil: Någon mer kategori?

Uppladdning: Några fler kriterier? Något som inte behövs?

Övrigt:

Editera/ta bort användare (editanv.php)

<http://qube3.thn.htu.se/~examen5/editanv.php>

Användare: Alla registrerade användare kommer att ligga i bokstavsordning. När man väljer en användare visas dess uppgifter fälten nedanför. Om man vill uppdatera/ändra några uppgifter gör man det genom att skriva i de nya uppgifterna och sedan trycka på knappen "Uppdatera". Vill man ta bort en användare trycker man helt enkelt på knappen "Ta bort" och svarar OK på frågan som dyker upp.

Synpunkter på hur användare listas i rullgardinsmenyn?

Övrigt:

Ta bort meddelanden (editmed.php)

<http://qube3.thn.htu.se/~examen5/editmed.php>

Här listas alla meddelanden som är inlagda i databasen. De listas under varandra, med rubrik, datum och användare och en del finns en "Ta bort"-knapp under varje meddelande. Vill man ta bort ett meddelande trycker man på "Ta bort"-knappen som hör till det meddelande man vill radera.

Saknas något?

Övrigt:

Registrera användare (reg.php)

<http://qube3.thn.htu.se/~examen5/reg.php>

Här kan administratörer gå in och lägga till en ny användare. Alla fält med *-tecken är obligatoriska och måste fyllas i. Vi har funderat på att skicka ett autogenerated e-mail till den nya användaren med dess användarnamn och lösenord. Om detta är svårt att lösa kommer administratören vara tvungen att kontakta den nya användaren och tala om användarnamn och lösenord.

Behövs det mer fält att fylla i här eller är det nog, t ex hemadress?

Befattning, **vilka olika befattningar ska finnas**, t ex städare, ekonomiansvarig?

Övrigt:

Meddelandesystem(medd.php)

<http://qube3.thn.htu.se/~examen5/medd.php>

På denna sida visas alla inlägg som finns i databasen. Om man vill begränsa sig och t ex bara titta på inlägg skrivna av administratörer väljer man detta i rullgardinsmenyn. Då kommer sidan att uppdatera sig och utvalda meddelanden att visas. Vill man så kan man även klicka på länken "Skriv nytt inlägg" och man skickas vidare till den sidan där man kan skriva nytt inlägg.

Behövs det fler kriterier som begränsar de meddelanden som visas, t ex när det är skrivet?

Synpunkter på utseendet av meddelandena:

Behövs länken "Skriv nytt inlägg" på något mer ställe, t ex i den svarta listan längst ner på sidan?

Övrigt:

Skriv inlägg (inlagg.php)

<http://qube3.thn.htu.se/~examen5/inlagg.php>

Här måste man fylla i rubrik och sedan skriver man ett litet meddelande, efter det klickar man på skicka knappen.

Några synpunkter på denna sida?

Övrigt:

Arkiv (arkiv.php)

<http://qube3.thn.htu.se/~examen5/arkiv.php>

Filarkivet, här kommer era olika mallar, scheman o s v ligga som länkar. Här kan man begränsa de olika filerna man vill se genom att välja vilken kategori de ska tillhöra. Detta gör man genom att trycka på rullgardningsmenyn och sedan trycka på knappen "Hämta filer"

Kategorierna vi har tänkt ha med:

Nyhetsbrev- alla nyhetsbrev med det senaste längst upp

Mallar- te x brevmallar

Scheman-

Möten- mötesscheman och protokoll

Övrigt- sådant som inte passar in på de andra...

Ska det vara fler/färre rubriker, i så fall vilka?

Behövs det fler avgränsningar än kategori, t ex datum när filen laddades upp?

Synpunkter?

Övrigt?

Ladda upp fil (upload.php)

<http://qube3.thn.htu.se/~examen5/upload.php>

Vill man skicka upp en fil till databasarkivet trycker man på knappen "Browse/Bläddra" och kan nu bläddra i sina kataloger på sin dator, och välja den filen man vill ladda upp. Filbeskrivningen är obligatorisk och fylls i fältet. Sedan väljer man vilken kategori filen ska tillhöra, och trycker på knappen "Skicka". Kategorierna kommer vara samma som på sidan <http://qube3.thn.htu.se/~examen5/arkiv.php>, d v s kommer att anpassas efter era synpunkter.

Synpunkter?

Övrigt:

Allmänna synpunkter på sidorna

Här kan du skriva allmänna synpunkter eller frågor som du har. Detta kan vara t ex det generella gränssnittet (utseendet). Är det svårt att navigera på de olika sidorna (svårt att hitta)? Är ifyllningsfälten otydliga? Alla åsikter som du har är värdefulla för oss för att kunna anpassa systemet efter just er som ska använda det.

Appendix 6

Feedback-svar till prototyp

Namn: Fredrik, Andrea, Mari.

Inloggning (index.php) <http://qube3.thn.htu.se/~examen5/>

På denna sidan skriver användaren in sitt användarnamn och lösenord och klickar på "logga in"-knappen och skickas automatiskt till startsidan. Är det en användare utan administratörsrättigheter kommer startsidan att innehålla fyra länkar. Är det en användare med administrationsrättigheter, kommer även administratörlänkarna upp.

Har ni några synpunkter på den här sidan, t ex design, användbarhet, rubriker?

Svar: Knappar eller någon ruta att fylla i för vilken kategori av personal man tillhör, eller styrs det via användare uppgifterna.

Exempel på kategorier av personalgrupper: Städare, Resurser, Arbetsledare, administrativ personal. Observera att nyttjandet av sidorna från städpersonalen förmodligen är liten.

Startsida (start.php) <http://qube3.thn.htu.se/~examen5/start.php>

På denna sidan visas länkar beroende på vad man är inloggad som.

Specifika länkar för administratörer:

Ta bort filer

Editera användare

Ta bort meddelanden

Registrera användare

Länkar för alla, inklusive administratören:

Meddelandesystemet

-Skriv nytt meddelande

Arkiv

-Ladda upp fil - Skicka upp en fil, t ex protokoll, nyhetsbrev...

Administratörlänkar

Saknas det något här? Något mer som endast administratören ska se?

Svar: Går det någonstans att få fram statistik som visar vilka som nyttjar sidorna?

Länkar

Länkar som alla ska se, synpunkter?

Svar: Länkarna "Meddelandesystem" och "skriv nytt meddelande" går inte de att få under samma länk? För om någon ställer en fråga under meddelanden så är det väl bra om den som svarar kan göra det på samma sida, som meddelande systemet finns på.

Går det att välja vilken kategori av personal som kan se respektive meddelande? Det kan ju vara så att jag skriver ett meddelande som enbart gäller arbetsledare, och då skall enbart de se det meddelandet. Likaväl som man skall kunna välja alla.

Finns det något arkiv eller liknande där gamla meddelanden sparas? Är det styrning på meddelande för respektive grupp?

Går det att exempelvis styra olika protokoll för respektive användaregrupp?

Kan man lägga de klickbara rubrikerna mer markerade exempelvis som knappar?

Behövs det något mer bland länkarna längst ned i det svarta fältet?

Svar: **Det räcker till att börja med.**

Ta bort filer (editfil.php) <http://qube3.thn.htu.se/~examen5/editfil.php>

Här kommer man kunna välja vilken kategori filen man vill ta bort ska höra till, och när den lades upp. Efter man valt det, trycker man på knappen "hämta fil". Filerna kommer hamna under varandra med en ta bort knapp under varje fil, för att enkelt radera den man vill.

Sökning:

Fil: Någon mer kategori?

Svar: **Ok till att börja med.**

Uppladdning: Några fler kriterier? Något som inte behövs?

Svar: **Det är ok.**

Editera/ta bort användare (editanv.php)

<http://qube3.thn.htu.se/~examen5/editanv.php>

Användare: Alla registrerade användare kommer ligga i bokstavsordning. När man väljer en användare visas dess uppgifter fälten nedanför. Om man vill uppdatera/ändra några uppgifter gör man det genom att skriva i de ny och sedan trycka på knappen "Uppdatera". Vill man ta bort en användare trycker man helt enkelt på knappen "Ta bort" och svarar OK på frågan som dyker upp.

Övrigt:

Svar: Det bör ligga med en registrering för vilken användare grupp man tillhör.

Jag förmodar också att det enbart är systemadministratören som kan lägga till eller ta bort användare.

Ta bort meddelanden (editmed.php)

<http://qube3.thn.htu.se/~examen5/editmed.php>

Här listas all meddelanden som är inlagda i databasen. De listas under varandra, med rubrik, datum och användare och en det finns en "Ta bort"-knapp under varje meddelande. Vill man ta bort ett meddelande trycker man på "Ta bort"-knappen som hör till det meddelande man vill radera.

Övrigt: Går detta också att styra efter respektive användargrupp? (inte nödvändigt). Eller kan man spara vissa meddelanden som man vill ha kvar någonstans?

Registrera användare (reg.php)

<http://qube3.thn.htu.se/~examen5/reg.php>

Här kan administratörer gå in och lägga till en ny användare. Alla fält med *-tecken är obligatoriska och måste fyllas i. Vi har funderat på att skicka ett autogenererat e-mail till den nya användaren med dess användarnamn och lösenord. Om detta är svårt att lösa kommer administratören vara tvungen att kontakta den nya användaren och tala om användarnamn och lösenord.

**Behövs det mer fält att fylla i här eller är det nog, t ex hemadress?
Det räcker bra.**

**Befattning, vilka olika befattningar ska finnas, tex städare,
ekonomiansvarig?**

**Städledare ändras till arbetsledare, sedan skall ytterligare en
befattning in "resurser".**

Meddelandesystem (medd.php)

<http://qube3.thn.htu.se/~examen5/medd.php>

På denna sida visas alla inlägg som finns i databasen. Om man vill begränsa sig och t ex bara titta på inlägg skrivna av administratörer väljer man detta i rullgardins-menyn. Då kommer sidan att uppdatera sig och utvalda meddelanden att visas. Vill man så kan man även klicka på länken "Skriv nytt inlägg" och man skickas vidare till den sidan där man kan skriva nytt inlägg.

**Behövs det fler kriterier som begränsar de meddelanden som visas,
t ex när det är skrivet?**

**Svar: Det skall fungera så att om man är inloggad som exempelvis
resurs, så skall man enbart se de inlägg som gäller denna
personalkategori eller de som gäller alla. Det måste vara styrt så
att man inte kan komma åt andra grupperns meddelanden.**

Synpunkter på utseendet av meddelandena:

Kan man under ett meddelande ha med något att trycka på om man vill lämna ett svar?

Behövs länken "Skriv nytt inlägg" på något mer ställe, tex i den svarta listan längst ner på sidan? Svar: Tror det är bra i den svarta listan.

Skriv inlägg (inlägg.php) <http://qube3.thn.htu.se/~examen5/inlagg.php>

Här måste man fylla i rubrik och sedan skriver man ett litet meddelande, efter det klickar man på skicka knappen.

Några synpunkter på denna sidan? Man bör kunna välja vilken personalkategori som meddelandet skall visas för, alternativt alla.

Arkiv (arkiv.php) <http://qube3.thn.htu.se/~examen5/arkiv.php>

Filarkivet, här kommer era olika mallar, scheman osv ligga som länkar. Här kan man begränsa de olika filerna man vill se genom att välja vilken kategori de ska tillhöra. Detta gör man genom att trycka på rullgardinsmenyn och sedan trycka på knappen "Hämta filer"

Kategorierna vi har tänkt ha med:

Nyhetsbrev- alla nyhetsbrev med med det senaste längst upp

Mallar- te x brevmallar

Scheman-

Möten- mötesscheman och protokoll

Övrigt- sådant som inte passar in på de andra...

Ska det vara fler/färre rubriker, i så fall vilka? Svar: Det är bra.

Behövs det fler avgränsningar än kategori, tex datum när filen laddades upp?

Skulle i så fall vara datum.

Synpunkter?

Är det avgränsat så att inte alla användare kommer åt exempelvis protokoll mm?

Ladda upp fil (upload.php) <http://qube3.thn.htu.se/~examen5/upload.php>

Vill man skicka upp en fil till databesarkivet trycker man på knappen "Browse/Bläddra" och kan nu bläddra i sina kataloger på sin dator, och välja den filen man vill ladda upp. Filbeskrivningen är obligatorisk och fylls i fältet. Sedan väljer man vilken kategori filen ska tillhöra, och trycker på knappen "Skicka". Kategorierna kommer vara samma som på sidan <http://qube3.thn.htu.se/~examen5/arkiv.php>, dvs kommer att anpassas efter era synpunkter.

Allmänna synpunkter på sidorna

Här kan du skriva allmänna synpunkter eller frågor som du har. Detta kan vara tex det generella gränssnittet (utseendet). Är det svårt att navigera på de olika sidorna (svårt att hitta)? Är ifyllnings-fälten otydliga? Alla åsikter som du har är värdefulla för oss för att kunna anpassa systemet efter just er som ska använda det.

Klickbara rubriker och rader bör vara utformade så att man ser att de är klickbara.

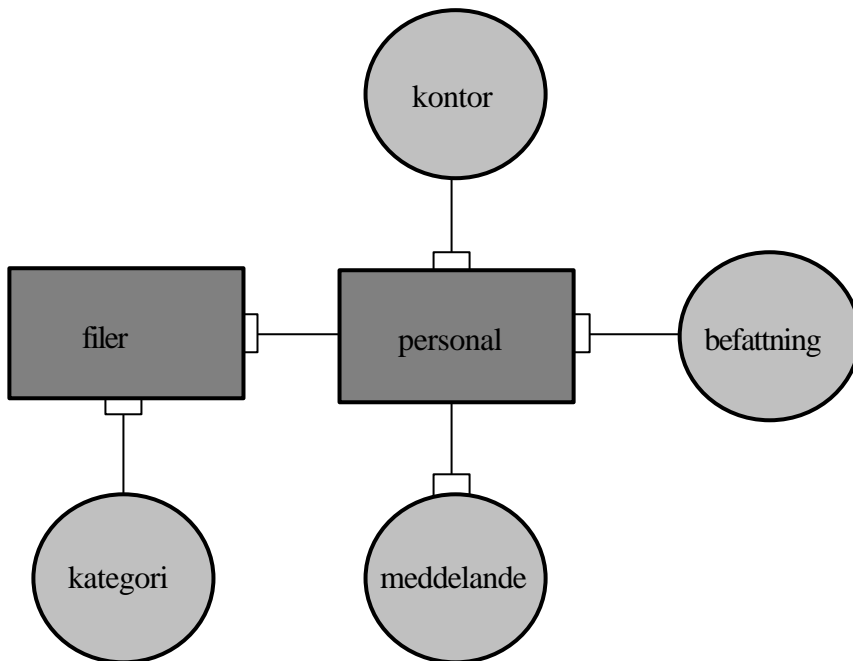
**Det är bara att berömma Er, med vissa justeringar så är det kanon!
Om vi varit för oklara i våra synpunkter så får Ni fråga igen?**

Appendix 7

Databasdesign

Innehåll:	Sid:
Objekttyper med relationer	2
Objekttermkatalog	3
Termkatalog	5
Normalisering (Logiska poster)	6
Datamodellering och volymberäkning	7

Objekttyper med relationer



Egenskaper

kontor:

*IDkontor
kontor

befattning:

*IDbefattning
befattning

kategori:

*IDkategori
kategori

personal:

*IDpersonal
IDpersonal
IDbefattning
losen
epost
telnr
nickname
fnamn
enamn

meddelanden:

*IDmeddelande
IDpersonal
rubrik
meddelande
meddattid
mbefattning

filer:

*IDfil
IDpersonal
IDkategori
filnamn
storlek
mimetyp
binardata
fildatum
beskrivning
fbefattning

Objektermkatalog

Objekttyp personal

Beskrivning: Anställda på företaget

Antal förekomster: 999

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
*IDpersonal	Anum	32
fnamn	Anum	50
enamn	Anum	50
nickname	Anum	25
losen	Anum	20
telnr	Anum	15
epost	Anum	50
IDkontor	Num	2
IDbefattning	Num	2

Objekttyp kontor

Beskrivning: Vilket kontor personalen tillhör

Antal förekomster: 99

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
*IDkontor	Num	2
kontor	Anum	25

Objekttyp befattning

Beskrivning: Personalens befattning

Antal förekomster: 99

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
*IDbefattning	Num	2
befattning	Anum	30

Objekttyp meddelanden

Beskrivning: Meddelande i meddelandesystemet

Antal förekomster: 99 999

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
*IDmeddelande	Num	5
meddelande	Anum	1024
meddattid	Anum	20
rubrik	Anum	50
IDpersonal	Anum	32
mbefattning	Anum	255

Objekttyp filer

Beskrivning: Uppladdade filer

Antal förekomster: 99 999

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
*IDfil	Num	5
filnamn	Anum	25
storlek	Anum	30
beskrivning	Anum	100
mimetyp	Anum	50
binardata	Anum	16 777 215
fildatum	Anum	20
IDkategori	Num	2
IDpersonal	Anum	32
fbefattning	Anum	255

Objekttyp kategori

Beskrivning: Filernas kategoribeteckning t ex nyhetsbrev

Antal förkomster: 99

<u>Termer:</u>	<u>Typ:</u>	<u>Antal tecken:</u>
-----------------------	--------------------	-----------------------------

*IDkategori	Num	2
kategori	Anum	50

Termkatalog

<u>Termnamn</u>	<u>Benämning</u>	<u>Typ</u>	<u>Tecken</u>
IDpersonal	Unikt nr som identifierar varje användare	Anum	32
IDbefattning	Unikt nr som identifierar personalens befattning	Num	2
IDmeddelande	Unikt nr som identifierar varje meddelande	Num	5
IDfil	Unikt nr som identifierar varje uppladdad fil	Num	5
IDkontor	Unikt nr som identifierar varje enskilt kontor	Num	2
IDkategori	Unikt nr som identifierar filkategori	Num	2
fnamn	Förnamn på användare	Anum	50
enamn	Efternamn på användare	Anum	50
nickname	Användarnamn	Anum	25
losen	Lösenord för inloggning	Anum	20
telnr	Telefonnr. till användare	Anum	15
epost	Epost till användare	Anum	50
kontor	Kontorsort	Anum	25
filnamn	Namnet på den uppladdade filen	Anum	25
storlek	Storlek på filen	Anum	30
beskrivning	Beskrivning av filen	Anum	100
mimetyf	Mimetyf på filen	Anum	50
binardata	Lagring av filen	Anum	16 777 215
fildatum	Datum när filen laddades upp	Anum	20
meddelande	Det skrivna meddelandet	Anum	1024
meddattid	Datum när meddelandet skrevs	Anum	20
rubrik	Rubrik till meddelandet	Anum	50

befattning	Användarens befattning	Anum	30
kategori	Filkategori	Anum	50
mbefattning	Vilken befattning personalen som skrivit ett meddelande har	Anum	255
fbefattning	Vilken befattning personalen som laddat upp en fil har	Anum	255

Normalisering (Logiska poster)

Personal: IDpersonal, M / fnamn, 1; enamn, 1; losen, 1; epost, 1; telnr, 1; nickname, 1; IDkontor, 1; IDbefattning, 1;

Kontor: IDkontor, M / kontor, 1;

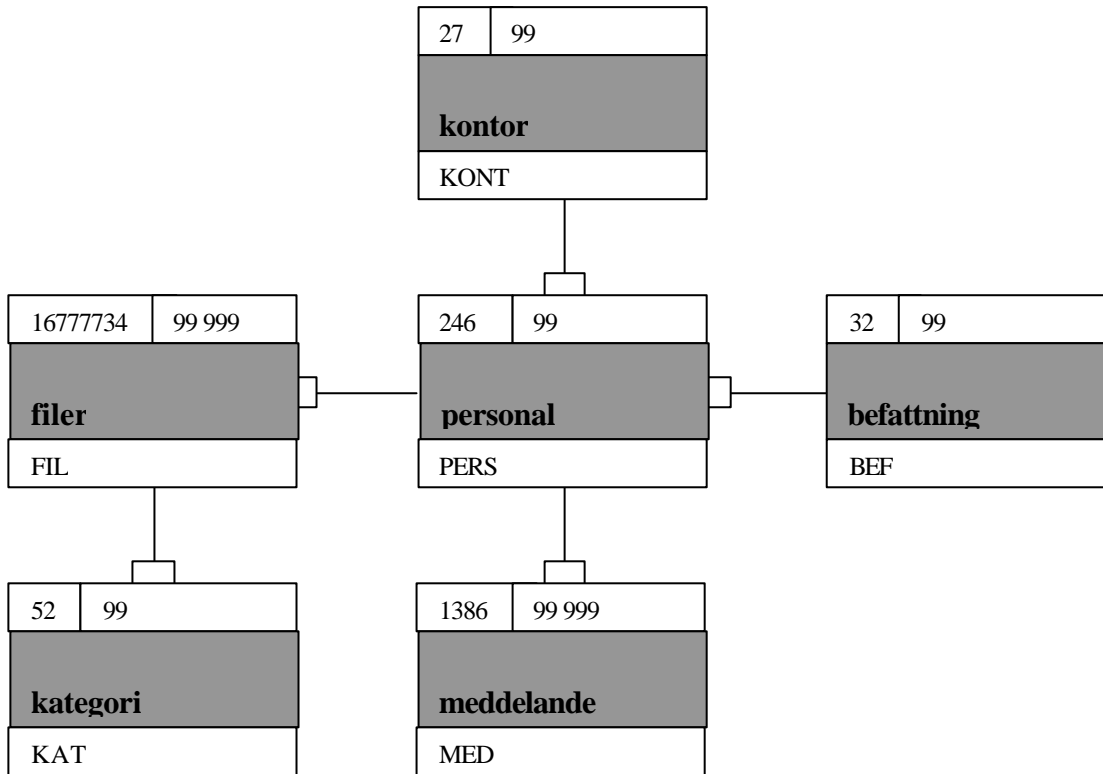
Befattning: IDbefattning, M / befattning, 1;

Meddelanden: IDmeddelande, M / meddattid, 1; meddelande, 1; rubrik, 1; IDpersonal, 1; mbefattning, 1;

Filer: IDfil, M / beskrivning, 1; IDkategori, 1; filnamn, 1; storlek, 1; mimetyp, 1; binardata, 1; fildatum, 1; IDpersonal 1; fbefattning, 1;

Kategori: IDkategori, M / kategori, 1;

Datamodellering och volymberäkning



Volymberäkning

	<u>Tecken</u>	<u>Poster</u>	<u>Totalt</u>
Personal	217	999	245 754
Kontor	27	99	2 673
Befattning	32	99	3 168
Meddelanden	1 386	99 999	138 598 614
Fil	16 777 734	99 999	1 677 756 622 266
Kategori	52	99	5 148

Lagringsvolym

≈1,68 miljarder tecken

Appendix 8

Grafisk presentation av databasen.

Alla tabeller i databasen:

Tabeller	Rader	Typ	Storlek
befattning	5	MyISAM	2,1 kB
filer	0	MyISAM	1,0 kB
kategori	0	MyISAM	1,0 kB
kontor	4	MyISAM	2,1 kB
meddelanden	1	MyISAM	2,0 kB
personal	7	MyISAM	2,5 kB

Tabell befattning:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDbefattning</u>	tinyint(2)		Nej		auto_increment
befattning	varchar(30)		Nej		

Tabell filer:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDfil</u>	mediumint(5)		Nej		auto_increment
IDkategori	tinyint(2)		Nej	0	
IDpersonal	varchar(32)		Nej	0	
filnamn	varchar(25)		Nej		
storlek	varchar(30)		Nej		
beskrivning	varchar(100)		Nej		
mimetyyp	varchar(50)		Nej		
binardata	mediumtext		Nej		
fildatum	datetime		Nej	0000-00-00	
fbefattning	varchar(255)		Nej		

Tabell kategori:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDkategori</u>	tinyint(2)		Nej		auto_increment
kategori	varchar(50)		Nej		

Tabell kontor:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDkontor</u>	tinyint(2)		Nej		auto_increment
kontor	varchar(25)		Nej		

Tabell meddelanden:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDmeddelande</u>	mediumint(5)		Nej		auto_increment
IDpersonal	varchar(32)		Nej	0	

meddelande	text	Nej	
meddattid	datetime	Nej	0000-00-00
rubrik	varchar(50)	Nej	
mbefattning	varchar(255)	Nej	

Tabell personal:

Fält	Typ	Attribut	Null	Standard	Extra
<u>IDpersonal</u>	varchar(32)		Nej		
IDkontor	tinyint(2)		Nej	0	
IDbefattning	tinyint(2)		Nej	0	
fnamn	varchar(50)		Nej		
enamn	varchar(50)		Nej		
nickname	varchar(25)		Nej		
losen	varchar(20)		Nej		
telnr	varchar(15)		Ja		
epost	varchar(50)		Ja		

Appendix 9

Programkod

PHP-FILER

addmsg.php

```
<?
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "addmsg.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');

// Instanciate the session class and the auth class
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH));

//include header of the page
include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_msg.class');

//create instanses of classes
$msg = new tis_msg;
$msg->debug=true;

//sends the message
if (isset($skicka))
{
    if($msg->validate($meddelande, $mbefattning))
    {
        $msg->add($meddelande, $mbefattning);
    }
    //Errormessage
    else
    {
        $error = $msg->error;
    }
}
}
?>
<!--Formtable-->
<form action="addmsg.php" method="post">
    <table border="0">
        <tr>
            <td width="40" rowspan="14"></td>
            <td>Rubrik </td>
            <td><input type="text" size="26" class="textfield"
name="meddelande[rubrik]"></td>
        </tr>
        <tr>
            <td valign="top">Meddelande </td>
            <td><textarea name="meddelande[meddelande]" rows="4"
cols="20"></textarea></td>
        </tr>
        <tr>
            <td height="0"></td>
        </tr>
        <tr>
            <td height="10"></td>
        </tr>
        <tr>
            <td></td>
            <td colspan="1">Meddelandet ska kunna läsas av</td>
        </tr>
        <tr>
            <td height="5"></td>
        </tr>
```



```

        <tr>
            <td rowspan="7"></td>
        <tr>
            <td><input type="checkbox" name="mbefattning[alla]"
value="alla">&nbsp;&nbsp;&nbsp;Alla</td>
        </tr>
        <tr>
            <td><input type="checkbox" name="mbefattning[administrator]"
value="administrator">&nbsp;&nbsp;&nbsp;Administratör</td>
        </tr>
        <tr>
            <td><input type="checkbox" name="mbefattning[administration]"
value="administration">&nbsp;&nbsp;&nbsp;Administration</td>
        </tr>
        <tr>
            <td><input type="checkbox" name="mbefattning[arbetsledare]"
value="arbetsledare">&nbsp;&nbsp;&nbsp;Arbetsledare</td>
        </tr>
        <tr>
            <td><input type="checkbox" name="mbefattning[resurs]"
value="resurs">&nbsp;&nbsp;&nbsp;Resurs</td>
        </tr>
        <tr>
            <td><input type="checkbox" name="mbefattning[stadare]"
value="stadare">&nbsp;&nbsp;&nbsp;Städare</td>
        </tr>
        <tr>
            <td colspan="2"><hr align="left" color="#ff9900" noShade size="1"></td>
        </tr>
        <tr>
            <td height="20"></td>
        </tr>
        <tr>
            <td></td>
            <td></td>
            <td align="right"><input class="button" type="submit" name="skicka"
value="Skicka">&nbsp;&nbsp;&nbsp;&nbsp;<input type="reset" class="button" name="rensa"
value="Rensa"></td>
        </tr>
        <tr>
            <td colspan="2" height="25"></td>
        </tr>
    </table>
</form>
<?php
    //include footer of the page
    include($_GAIA_WEBROOT_DIR.'footer.php');
    //ErrorMessage
    if($error)
    {
        foreach($error as $key => $value)
        {
            $error_msg.= $error[$key];
        }
        echo
        (
            <script language='JavaScript'>
                window.alert('".$error_msg."');
            </script>
            <noscript>Du måste fylla i alla fält!</noscript>
        ");
    }
    page_close();
?>

```

adduser.php

```

<?
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "adduser.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH));

//include header of the page
include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_user.class');
include(_GAIA_DIR.'tis_position.class');
include(_GAIA_DIR.'tis_office.class');

//create instanses of objects
$db = new db_gaia;
$usr = new tis_user;
$position = new tis_position;
$office = new tis_office;

//Add user to db
if (isset($skicka))
{
    if ($usr->validate($personal, $IDbefattning, $IDkontor))
    {
        $usr->add($personal, $IDbefattning, $IDkontor);

        //Update page
        unset($IDbefattning, $IDkontor, $personal);
    }
    //ErrorMessage
    else
    {
        $error = $usr->error;
    }
}
?>

<!--Formtable-->
<form action="adduser.php" method="post">
    <table border="0">
        <tr>
            <td colspan="2" height="15"></td>
        </tr>
        <tr>
            <td width="40" rowspan="15"></td>
            <td colspan="2">Fält markerade med * är obligatoriska och måste
fyllas i</td>
        </tr>
        <tr>
            <td height="20"></td>
        </tr>
        <tr>
            <td>Förnamn * </td>
            <td><input type="text" size="25" class="textfield" value='<?php echo
$personal[fnamn]; ?>' name="personal[fnamn]"></td>
        </tr>
        <tr>
            <td>Efternamn * </td>
            <td><input type="text" size="25" class="textfield" value='<?php echo
$personal[enamn]; ?>' name="personal[enamn]"></td>

```



```

include(_GAIA_DIR.'tis_file.class');

$category = new tis_category;
$file = new tis_file;

if(isset($tabort))
{
    // Converts id from one array to a value in another array
    $IDfil = array_keys($tabort);
    $file->remove($IDfil[0]);
}

if(isset($IDkategori))
{
    $file->get($IDkategori);
    $filer = $file->filer;
}
?>
<!--script for check if the file should be removed or not-->
<script language='JavaScript'>
function confirmDelete()
{
    window.event.returnValue=false;

    if(confirm ('Är du säker på att du vill ta bort filen?'))
    {
        window.event.returnValue=true;
    }
}
</script>
<!--Formtable-->
<form action="editfile.php" method="post">
    <table border="0">
        <tr>
            <td height="60"></td>
        </tr>
        <tr>
            <td width="40" rowspan="100"></td>
            <td>Filkategori</td>
            <td>
                <?php
                    //build drop-down-list with categories
                    $category->build_ddlb($IDkategori, true);
                ?>
            </td>
        </tr>
        <tr>
            <td></td>
        </tr>
        <tr>
            <td></td>
        </tr>
        <tr>
            <td height="30"></td>
        </tr>
        <tr>
            <td colspan="3"><hr align="left" color="#ff9900" noShade
size="1"></td>
        </tr>
        <?php
            $i=0;
            //show files with a remove button for each
            if(count($filer)>0)
            {

```


editmsg.php

```

<?
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "editmsg.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
// Instanciate the session class and the auth class
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH, "perm" =>
_GAIA_PERM));
include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_msg.class');

$msg = new tis_msg;

if(isset($rensa))
{
    // Converts id from one array to a value in another array
    $removeID = array_keys($rensa);
    $msg->remove($removeID[0]);
}
$msg->get($meddelande);
$meddelande = $msg->meddelande;
?>
<!--script for check if the message should be removed or not-->
<script language='JavaScript'>
function confirmDelete()
{
    window.event.returnValue=false;

    if(confirm ('Är du säker på att du vill ta bort meddelandet?'))
    {
        window.event.returnValue=true;
    }
}
</script>
<!--Formtable-->
<form action="editmsg.php" method="post">
    <table border="0">
        <tr>
            <td height="5"></td>
        </tr>
        <tr>
            <td width="40" rowspan="100">
                <td colspan="3"><hr align="left" color="#ff9900" noShade
size="1"></td>
            </tr>
            <?php
                $i=0;
                $storlek = sizeof($meddelande);
                while($i < $storlek)
                {
                    echo
                    ( "
                        <tr>
                            <td class='msgheadline'>".wordwrap($meddelande[$i][rubrik],
35, '<br>')."&nbsp;</td><td valign='bottom'
class='datetime'>". $meddelande[$i][meddattid]."</td>
                        </tr>
                        <tr>
                            <td height='4'></td>
                        </tr>
                    "
                )
            </td>
        </tr>
    </table>

```



```

        <tr>
            <td><i>".$meddelande[$i][fullname]."</i></td>
            <td align='right' valign='bottom' rowspan='2'><input
type='submit' onclick='return confirmDelete();' class='button'
name='rensa[ ".$meddelande[$i][IDmeddelande]."]' value='Ta bort'></td>
        </tr>
        <tr>
            <td class='datetime'>".$meddelande[$i][befatt]."</td>
        </tr>
        <tr>
            <td colspan='3'><hr align='left' color='#ff9900' noShade
SIZE='1'></td>
        </tr>
        ");
        $i++;
    }
?>
<tr>
    <td colspan="2" height="10"></td>
</tr>
</table>
</form>
<?
    include(_GAIA_WEBROOT_DIR.'footer.php');
    page_close();
?>

```

edituser.php

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "edituser.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
// Instantiate the session class and the auth class
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH));

include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_user.class');
include(_GAIA_DIR.'tis_position.class');
include(_GAIA_DIR.'tis_office.class');

//create new instanses of used objects
$db = new db_gaia;
$usr = new tis_user;
$position = new tis_position;
$office = new tis_office;

//show a user if the user is choosen
if(isset($anvandare) && !$supdatera)
{
    if(isset($anvandare))
    {
        $usr->get($anvandare);
        $personal=$usr->personal;
    }
    else
    {
        refresh("edituser.php");
    }
}
//remove a user

```



```

</form>
<?
//include the footer of the web-site
include("/home/sites/home/users/examen5/web/footer.php");
//ErrorMessage
if($error)
{
    foreach($error as $key => $value)
    {
        $error_msg.= $error[$key];
    }
    echo
    ( "
        <script language='JavaScript'>
            window.alert('".$error_msg."');
        </script>
        <noscript>Du måste fylla i alla fält!</noscript>
    " );
}
page_close();
?>

```

footer.php

```

<?php
echo
( "
    <td align='left' rowspan='100'><img src='./images/kvast.jpg' border='0'
width='100' alt=''></td>
</tr>
<tr>
");
if ($page == 'index.php' || $page == 'logout.php')
{
    echo
    ( "
        <td class='bottomlank' bgcolor='#000000' height='35' colspan='2'
align='center'></td>
    " );
}
else
{
    echo
    ( "
        <td class='bottomlank' fontcolor='#FFFFFF' bgcolor='#000000' height='35'
colspan='2' align='center'><font color='#FFFFFF'><a href='./index.php'
class='bottom'><strong>Startsida</a> ~ <a href='./showmsg.php'
class='bottom'>Meddelandesystem</a> ~ <a href='./archive.php'
class='bottom'>Filarkiv</a> ~ <a href='./logout.php' class='bottom'>Logga
ut</strong></font></td>
    " );
}
echo
( "
        </tr>
    </form>
</table>
</td>
</tr>
</table>
</body>
</html>
");

```

?>

gaia.conf

```
<?php
/*****
 *
 * GAIA.CONF - Configuration Options
 *
 *****/
 *
 * See ./doc/gaia.conf.html for documentation.
 *
 * $Revision: 1.9 $
 * $Author: carro$
 * $Date: 2001/10/12 12:08:39 $
 *
 * Copyright (c) 2002 Magnus Jonsson. All rights reserved.
 *
 *****/

// SERVER SETTINGS
define("_GAIA_SMTMP_SERVER", "localhost");
define("_GAIA_APACHE_BIN_DIR", "/www/bin/");
define("_GAIA_APACHE_LOG_DIR", "/www/logs/");

// DATABASE SETTINGS
define("_GAIA_DB_DRIVER", "mysql");
define("_GAIA_DB_SERVER", "localhost:3306");
define("_GAIA_DB_DATABASE", "examen5");
define("_GAIA_DB_USERNAME", "examen5");
define("_GAIA_DB_PASSWORD", "FZWMMG");
$XFC_DEFAULT_DB_CLASS="db_gaia";

// PATH SETTINGS
define("_GAIA_DIR", "/home/sites/home/users/examen5/gaia/");
define("_GAIA_URL", "http://qube3.thn.htu.se/~examen5/");
define("_GAIA_TMP_DIR", "/tmp/");
define("_GAIA_WEBROOT_DIR", "/home/sites/home/users/examen5/web/");

// SESSION CONSTANTS
define("_GAIA_SESS", "gaias");
define("_GAIA_AUTH", "gaiaa");
define("_GAIA_PERM", "gaiap");
define("_GAIA_USER", "gaiau");

// REGIONAL SETTINGS
define("_GAIA_DECIMAL_SYMBOL", ",");
define("_GAIA_DIGIT_GROUPING_SYMBOL", ".");
define("_GAIA_TIME_STYLE", "H:i:s");
define("_GAIA_DATETIME_STYLE", "Y-m-d H:i");
define("_GAIA_DATE_STYLE", "Y-m-d");
define("_GAIA_DATE_SEPARATOR", "-");
define("_GAIA_LONG_DATE_STYLE", "F j, Y, G:i");
define("_GAIA_TIMEZONE_MODIFIER", 0);
define('_GAIA_DECIMAL_DIGITS', 2);
define('_GAIA_CURRENCY', 'SEK');
define('_GAIA_LANGUAGE', 'en');

define('_GAIA_DATE', date(_GAIA_DATE_STYLE, (time()-5184000)));
```

```

define('_GAIA_CURRENT_DATE',date(_GAIA_DATE_STYLE,(time()+_GAIA_TIMEZONE_MODIFIER
));
define('_GAIA_CURRENT_DATETIME',date(_GAIA_DATETIME_STYLE,(time()+_GAIA_TIMEZONE_
MODIFIER)));
// LAYOUT CONSTANTS
define("_GAIA_STYLESHEET","/default.css");

// INCLUDE INITIALIZATION CODE
include(_GAIA_DIR.'core/gaia_init.php');
?>
-----
-----

```

gaia_login.inc

```

<?php
/**
 * GAIA_INIT.PHP
 *
 * Copyright (c) 2002 Magnus Jonsson. All rights reserved.
 *
 * @author Initial version by Magnus Jonsson <magnusj@phrenetic.net>
 * @version $Revision: 1.2 $, $Date: 2002/02/05 14:00:13 $
 */

//includes header of the page
$page = "index.php";
include(_GAIA_WEBROOT_DIR.'header.php');

    // *** USERNAME TEXTBOX ***
    if (isset($this->auth["uname"]) || isset($submit))
    {
        $username_value = $this->auth["uname"];
    }
    else
    {
        $username_value = "";
    }
?>
<!--Formtable-->
<table border="0" >
    <form name='' method='post' action='./index.php' target='_self'
onsubmit="return login_Validator(this)">
        <tr>
            <td height="60">
                </td>
            </tr>
            <tr>
                <td width="40" rowspan="5"></td>
                <td>Användarnamn</td>
                <td><input type="text" name='username' class="textfield"
maxlength='20' size='18' name="username"></td>
            </tr>
            <tr>
                <td>Lösenord </td>
                <td><input type="password" class="textfield" name="password" value=""
maxlength='20' size='18'></td>
            </tr>
            <tr>
                <td colspan="2"><hr align="left" color="#ff9900" noShade
size="1"></td>
            </tr>
            <tr>
                <td></td>
            </tr>
            <tr>
                <td></td>
            </tr>
        </form>
    </table>

```

```

        <td><input name="submit" value="Logga in" type="submit"
class="button"></td>
    </tr>
    <tr>
        <td colspan="2"></td>
    </tr>
    <tr>
        <td colspan="2" height="200"></td>
    </tr>
</form>
<script language='javascript'>
<!--
    function login_Validator(f)
    {
        if (f.elements[0].value.length < 1)
        {
            alert("Du måste ange ditt användarnamn för att logga in!");
            f.elements[0].focus();
            return(false);
        }
        if (f.elements[1].value.length < 1)
        {
            alert("Du måste ange ditt lösenord!");
            f.elements[1].focus();
            return(false);
        }
    }
    //-->
</script>
</table>
<?php
    global $username;
    if ( isset($username) )
    { ?>
        <!-- failed login code -->
        <p>
        <table>
            <tr>
                <td class="error">Antingen är ditt användarnamn '<?php print $this-
>auth["uname"]; ?>' eller ditt lösenord felaktigt!<br> Försök igen.</td>
            </tr>
        </table>

<?php } ?>

<script language="JavaScript">
<!--
    if (document.forms[0][0].value != '')
    {
        document.forms[0][1].focus();
    }
    else
    {
        document.forms[0][0].focus();
    }
    -->
</script>
<?php
//includes footer of the page
include($_GAIA_WEBROOT_DIR.'footer.php');
?>
-----
getfil.php

```

```
<?php
//This page uses functionality from the framework GAIA.

    include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
    include(_GAIA_DIR.'tis_file.class');
    $fil = new tis_file;
    $fil->get_file($id);
?>
```

header.php

```
<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net and www.webdesignskolan.com*/

/*$popup- shows a helpwindow
  $rubrik- headline, depending on which page you are on*/

if ($page=="addmsg.php")
{
    $popup = './help/addmsgghelp.html';
    $rubrik = 'Lägg till meddelande';
}
if ($page=="index.php")
{
    $popup = './help/indexhelp.html';
    $rubrik = 'Logga in';
}
if ($page=="uploadfile.php")
{
    $popup = './help/uploadfilehelp.html';
    $rubrik = 'Ladda upp fil';
}
if ($page=="adduser.php")
{
    $rubrik = 'Lägg till användare';
}
if ($page=="archive.php")
{
    $popup = './help/archivehelp.html';
    $rubrik = 'Filarkivet';
}
if ($page=="editfile.php")
{
    $rubrik = 'Ta bort fil';
}
if ($page=="editmsg.php")
{
    $rubrik = 'Ta bort meddelande';
}
if ($page=="showmsg.php")
{
    $popup = './help/showmsgghelp.html';
    $rubrik = 'Meddelandesystemet';
}
if ($page=="start.php")
{
    $popup = './help/starthelp.html';
    $rubrik = 'Länkar';
}
if ($page=="edituser.php")
{

```



```

    $rubrik = 'Ta bort/editera användare';
}
if ($page=="logout.php")
{
    $rubrik = 'Utloggad';
}

echo
(
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<!--head on all pages-->
<html>
    <head>
        <link rel='stylesheet' href='stylesheet.css'>
        <title>T.I.S - Trasan Information System</title>
        <script language='JavaScript' type='text/javascript'>
            <!-- Begin Webbdesignskolan.com
            function help()
            {
                attribute=window.open("%. $popup.", 'poppage', 'toolbars=0,
scrollbars=0, location=0, statusbars=0, menubars=0, resizable=0, width=300,
height=350 left = 50, top = 100');
            }
            // End -->
        </script>
    </head>
    <body bgcolor='#ff9900'>
        <table height='100%' align='center'>
            <tr>
                <td>
                    <table style='border: 1 solid #000000' border='0'
cellpadding='0' cellspacing='0' bgcolor='#FFFFFF' >
                        <tr>
                            <td colspan='2'><img src='./images/head.jpg'
alt=''></td>
                        </tr>
                        <tr>
                            <td>
                                <!--Headlinetable-->
                                <table border='0'>
                                    <tr>
                                        <td height='10'></td>
                                    </tr>
                                    <tr>
                                        <td width='40' rowspan='3' ></td>
                                        <td class='headline'>".$rubrik."<td>
                                    </tr>
                                    <tr>
                                        <td><hr align='left' color='#000000'
width='350' noShade SIZE='1'></td>
                                    </tr>
                                </table>
                            </td>
                        </tr>
                        <tr>
                            <td>
                                <!--Headlinetable-->
                                <table border='0'>
                                    <tr>
                                        <td height='10'></td>
                                    </tr>
                                    <tr>
                                        <td width='40' rowspan='3' ></td>
                                        <td class='headline'>".$rubrik."<td>
                                    </tr>
                                    <tr>
                                        <td><hr align='left' color='#000000'
width='350' noShade SIZE='1'></td>
                                    </tr>
                                </table>
                            </td>
                        </tr>
                    </table>
                </td>
            </tr>
        </table>
    </body>
</html>
);
if($page=='archive.php' || $page=='index.php' ||
$page=='addmsg.php' || $page=='showmsg.php' || $page=='uploadfile.php' ||
$page=='start.php')
{
    echo
    (
        <tr>
            <td><a href=\"javascript:help()\"
onmouseover=\"document.bild1.src='./images/help2.jpg'; return true\"
onmouseout=\"document.bild1.src='./images/help1.jpg'; return true\"><img
src='./images/help1.jpg' name='bild1' alt='Hjälp' border=0></a></td>
        </tr>
    )
}

```



```
        <tr>
            <td colspan='2' height='170'></td>
        </tr>");
    }
    else
    {
        echo("
        <tr>
            <td colspan='2' height='117'></td>
        </tr>");
    }
    echo("</table>");
    include(_GAIA_WEBROOT_DIR.'footer.php');
    page_close();
?>
```

logout.php

```
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

<?php
    $page="logout.php";

    include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
    // Include page functionality
    page_open(array("sess"=>_GAIA_SESS, "auth"=>_GAIA_AUTH, "perm"=>_GAIA_PERM));

    $sess->delete();
    $auth->logout();

    include(_GAIA_WEBROOT_DIR.'header.php');
?>
<table border="0" align='center'>
    <tr height="130"><td></td></tr>
    <tr>
        <td class='msgheadline'>Du är utloggad från T.I.S</td>
    </tr>
    <tr>
        <td><a href="./index.php">Klicka här för att logga in på nytt!</a></td>
    </tr>
    <tr height="230"><td></td></tr>
</table>
<?php
    include(_GAIA_WEBROOT_DIR.'footer.php');
    page_close();
?>
```

showmsg.php

```
<?
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "showmsg.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
// Instanciate the session class and the auth class
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH, "perm" =>
_GAIA_PERM));

include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_msg.class');
include(_GAIA_DIR.'tis_position.class');
```

```

    $msg = new tis_msg;
    $position = new tis_position;
    $msg->get($meddelande, true);
    $meddelande = $msg->meddelande;
?>
<!--Formtable-->
<form action="showmsg.php" method="post">
    <table border="0">
        <tr>
            <td width="40"></td>
            <td colspan="2"><a href="./addmsg.php" class="lankar">Skriv nytt
inlägg</a></td>
        </tr>
        <tr>
            <td height="30"></td>
        </tr>
        <tr>
            <td width="40" rowspan="100">
            </td>
        </tr>
        <tr>
            <td colspan="2"><hr align="left" color="#ff9900" noShade size="1"></td>
        </tr>
        <?php
            $i=0;
            //shows messages
            if(count($meddelande)>0)
            {
                foreach($meddelande as $i => $value )
                {
                    echo
                    ( "
                        <tr>
                            <td
width='180'class='msgheadline'>".$meddelande[$i][rubrik]."</td>
                            <td align='right'
class='datetime'>".$meddelande[$i][meddattid]."</td>
                        </tr>
                        <tr>
                            <td colspan='2'>".wordwrap($meddelande[$i][meddelande], 60,
'<br>')."</td>
                        </tr>
                        <tr height='5'><td colspan='2'></td></tr>
                        <tr>
                            <td colspan='2'
align='right'><i>".$meddelande[$i][fullname]."</i></td>
                        </tr>
                        <tr>
                            <td colspan='2' align='right'
class='datetime'>".$meddelande[$i][befatt]."</td>
                        </tr>
                        <tr>
                            <td colspan='2'><hr align='left' color='#ff9900' noShade
size='1'></td>
                        </tr>
                    " );
                    $i++;
                }
            }
?>
        <tr>
            <td colspan="2" height="10"></td>
        </tr>
    </table>

```

```

    </table>
</form>
<?
    include(_GAIA_WEBROOT_DIR.'footer.php');
    page_close();
?>

```

uploadfile.php

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

$page = "uploadfile.php";
include('/home/sites/home/users/examen5/gaia/conf/gaia.conf');
// Instantiate the session class and the auth class
page_open(array("sess" => _GAIA_SESS, "auth" => _GAIA_AUTH));

include(_GAIA_WEBROOT_DIR.'header.php');
include(_GAIA_DIR.'tis_file.class');
include(_GAIA_DIR.'tis_category.class');

$db = new db_gaia;
$file = new tis_file;
$category = new tis_category;

//sends the file to the db with this different types
if (isset($skicka))
{
    $filer[0][storlek] = $fil_size;
    $filer[0][mimetyf] = $fil_type;
    $filer[0][filnamn] = $fil_name;
    $filer[0][binardata] = $fil;

    //Upload a file to db
    if ($file->validate($filer, $fbefattning, $IDkategori))
    {
        $file->add($filer[0], $fbefattning, $IDkategori);
    }
    else
    {
        $error = $file->error;
    }
}
?>
<!--Formtable-->
<!--
<form enctype="multipart/form-data" action="uploadfile.php" method="post">
    <table border="0">
        <tr>
            <td width="40" rowspan="14"></td>
            <td><input type="hidden" name="MAX_FILE_SIZE" value="2097152">Välj fil
att ladda upp:</td>
        </tr>
        </tr>
            <td><input name="fil" type="file" SIZE=30></td>
        </tr>
        <tr>
            <td height="10"></td>
        </tr>
        <tr>
            <td>Filbeskrivning (beskriv vad det är för fil du ska ladda upp)</td>
        </tr>

```



```

        </script>
        <noscript>Du måste fylla i alla fält!</noscript>
    ");
}
page_close();
?>

```

STYLESHEET

stylesheet.css

```

//a lot of ideas on this page come from www.webdesignskolan.com

.bottomlink { font-family: Arial; font-size: 9 pt; color:#FFFFFF}
.lankar {font-family: Arial; font-size: 11pt}
.footer {font-family: Verdana; color: #000000; font-size: 8pt}
td { font-family: Arial; font-size: 9pt}
ul { list-style-type: circle}

.linkheadline { color:#000000; font-face:Arial; font-size:12pt; font-weight:700}
.headline {color:#FF3300; font-face:Arial; font-size:12pt; font-weight:700}
.button { background-color: #FFFFFF; border: #ff9900; border-style: solid;
border-top-width: 1px; border-right-width: 2px; border-bottom-width: 2px; border-
left-width: 1px}
.textfield { border: #ff9900; border-style: solid; border-top-width: 1px; border-
right-width: 1px; border-bottom-width: 1px; border-left-width: 1px}
.scroll {scrollbar-base-color: #003366; scrollbar-track-color: #003366;
scrollbar-arrow-color: #0099CC;}
.msg { border: none;}
.msgheadline {color:#FF3300; font-face:Arial; font-size:11pt; font-weight:600}
.datetime {color:#000000; font-face:Arial; font-size:8pt;}

a:link { text-decoration: underline; color: #FF6600}
a:visited { text-decoration: underline; color: #FF3300}
a:active { text-decoration: underline; color: #FF3300}
a:hover { text-decoration: underline; color: #FF6600}

a.bottom:link { text-decoration: none; color: #FFFFFF}
a.bottom:visited { text-decoration: none; color: #FFFFFF}
a.bottom:active { text-decoration: none; color: #FFFFFF}
a.bottom:hover { text-decoration: underline; color: #FFFFFF}

```

KLASS-FILER

tis_category.class

```

<?php

/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

// Class with functions for handling categories
// The category class inherits the functions from class db_gaia
class tis_category extends db_gaia
{
    var $debug    = false;

    // Function for build a drop-down-list with all the categories in
    function build_ddlb($kategori=false, $submit=false)
    {
        $query = "SELECT * FROM kategori";

        // If something goes wrong the sql-query is displayed on the screen

```

```

if($this->debug)
{
    echo $query;
}

// Function for connecting to db and execute the SQL-query
$this->query($query);

// Build the drop-down-list
if($submit==true)
{
    echo '<select name="IDkategori" style="color: #CC6600"
onchange="submit()">';
}
else
{
    echo '<select name="IDkategori" style="color: #CC6600">';
}

echo '<option>-Kategori-</option>';

while($this->next_record())
{
    if ($kategori == $this->f(IDkategori))
    {
        echo '<option value="'. $this->f(IDkategori).'" selected>'. $this-
>f(kategori). '</option>';
    }
    else
    {
        echo '<option value="'. $this->f(IDkategori).'">'. $this-
>f(kategori). '</option>';
    }
}
echo '</select>';
}
}
?>

```

tis_file.class

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

// Class with functions for handling files, like add and remove
// The file class inherits the functions from class db_gaia
class tis_file extends db_gaia
{
    var $debug    = false;
    var $error = array();

    function validate($filer, $fbefattning, $IDkategori)
    {
        //Control that all mandatory fields are filled in
        if (strlen(trim($filer[0][beskrivning]))<1 || strlen($filer[0][filnamn])<1)
        {
            $this->error[] = "Du måste fylla i alla obligatoriska fält ";
        }

        if (count($fbefattning)<1)
        {

```



```
        $this->error[] = "Du har glömt att ange vem som ska få tillgång till
filen! ";

    }
    if (!is_numeric($IDkategori))
    {
        $this->error[] = "Du måste ange en filkategori!" ;
    }

    if (count($this->error)>0)
    {
        return false;
    }
    return true;
}

//Function for adding files to the database
function add($filer, $fbefattning, $IDkategori)
{
    global $auth;
    // Check if it is a file
    if(is_uploaded_file($filer[binardata]))
    {
        // Open file and put it in variable $fd
        $fd = fopen ($filer[binardata], "rb");

        // base64_encode - set escapecharacters
        $filer[binardata] = base64_encode(fread($fd, filesize
($filer[binardata])));

        // Close file
        fclose ($fd);
    }

    /* Choose which string to put in the variable $befattning
depending on position in the company*/
    $befattning = "";
    if (isset($fbefattning[stadare]))
    {
        $befattning.= "1,";
    }
    if (isset($fbefattning[resurs]))
    {
        $befattning.= "2,";
    }
    if (isset($fbefattning[arbetsledare]))
    {
        $befattning.= "3,";
    }
    if (isset($fbefattning[administration]))
    {
        $befattning.= "4,";
    }
    if (isset($fbefattning[administrator]))
    {
        $befattning.= "5,";
    }
    if (isset($fbefattning[alla]))
    {
        $befattning = "1,2,3,4,5,";
    }

    //addslashes - quote string with slashes
    $filer[beskrivning] = addslashes($filer[beskrivning]);
}
```

```

// Removes the comma in the end of the string
$befattning = substr($befattning, 0, -1);

$IDpersonal = $auth->auth[uid];

$query = "INSERT INTO filer (IDkategori, IDpersonal, filnamn, storlek,
beskrivning, mimetyyp, binardata, fbefattning, fildatum) VALUES('$IDkategori',
'$IDpersonal', '$filer[filnamn]', '$filer[storlek]', '$filer[beskrivning]',
'$filer[mimetyyp]', '$filer[binardata]', '$befattning',
'".$_GAIA_CURRENT_DATETIME."')";

// If something goes wrong the sql-query is displayed on the screen
if ($this->debug)
{
    echo $query;
}

// Function for connecting to db and execute the SQL-query
$this->query($query);
//refresh("showmsg.php");
}

// Function for delete files from db
function remove($IDfil)
{
    // is_numeric - check for a numeric value
    if(is_numeric($IDfil))
    {
        $query = "DELETE FROM filer WHERE IDfil='$IDfil'";
        $this->query($query);
    }
}

// Function for get a file from the db
function get_file($id)
{
    if (is_numeric($id))
    {
        $query = "SELECT mimetyyp, filnamn, binardata, storlek FROM filer WHERE
IDfil = $id";
        $this->query($query);
        $this->next_record();
        header("Content-Disposition: Filename=".$this->f(filnamn));
        header("Content-type: ".$this->f(mimetyyp));
        header("Content-length: ".$this->f(storlek));
        header("Pragma: no-cache");
        header("Expires: 0");
        echo base64_decode($this->f(binardata));
    }
}

// Function for display/show files
function get($IDkategori)
{
    global $auth;
    if(is_numeric($IDkategori))
    {
        $query = "SELECT beskrivning, fbefattning, IDfil, fildatum, filnamn FROM
filer WHERE IDkategori=$IDkategori ORDER BY fildatum DESC";
        $this->query($query);
        $i = 0;

        // Loops through all files putting them into an array
    }
}

```

```

while($this->next_record())
{
    $permission = strstr($this->f(fbefattning), $auth-
>auth[IDbefattning]);

    if($permission)
    {
        //htmlentities - convert both double and single quotes
        //stripslashes - Un-quote string
        $this->filer[$i][IDfil] = $this->f(IDfil);
        $this->filer[$i][filnamn] = $this->f(filnamn);
        $this->filer[$i][beskrivning] = htmlentities(stripslashes($this-
>f(beskrivning)), ENT_QUOTES);
        $this->filer[$i][fildatum] = $this->f(fildatum);
        $i++;
    }
}
}
}
}
?>

```

tis_msg.class

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

// Class with functions for handling messages, like add and remove
// The message class inherits the functions from class db_gaia
class tis_msg extends db_gaia
{
    var $debug      = false;

    //array for error messages
    var $error      = array();

    function validate($meddelande, $mbefattning)
    {
        //trim - Removes whitespaces from beginning and end of the string
        //strlen - checks if the variable contains more than 1 character
        if (strlen(trim($meddelande[rubrik]))<1 ||
strlen(trim($meddelande[meddelande]))<1 || strlen($mbefattning)<1)
        {
            //$this->error[] points to it's own class
            $this->error[] = " Du måste fylla i alla obligatoriska fält ";
        }
        //Check if the array contains anything
        if(count($this->error)>0)
        {
            return false;
        }
        return true;
    }

    // Function for add messages to the db
    function add($meddelande, $mbefattning)
    {
        global $auth;

        /* Choose which string to put in the variable $befattning
        depending on position in the company*/
        $befattning = "";
    }
}

```

```

if (isset($mbefattning[stadare]))
{
    $befattning.= "1,";
}
if (isset($mbefattning[resurs]))
{
    $befattning.= "2,";
}
if (isset($mbefattning[arbetsledare]))
{
    $befattning.= "3,";
}
if (isset($mbefattning[administration]))
{
    $befattning.= "4,";
}
if (isset($mbefattning[administrator]))
{
    $befattning.= "5,";
}
if (isset($mbefattning[alla]))
{
    $befattning = "1,2,3,4,5,";
}
else
{
    $befattning.= "5,";
}

//addslashes - quote string with slashes
$meddelande[rubrik] = addslashes($meddelande[rubrik]);
$meddelande[meddelande] = addslashes($meddelande[meddelande]);

// Removes the comma in the end of the string
$befattning = substr($befattning, 0, -1);

$IDpersonal = $auth->auth[uid];

$query = "INSERT INTO meddelanden (IDpersonal, meddelande, meddattid,
rubrik, mbefattning) VALUES('$IDpersonal', '$meddelande[meddelande]',
'".$_GAIA_CURRENT_DATETIME."', '$meddelande[rubrik]', '$befattning')";

// Function for connecting to db and execute the SQL-query
$this->query($query);

}
// Function for delete messages from the db
function remove($rensa)
{
    //is_numeric - checks if the value is a numeric value
    if(is_numeric($rensa))
    {
        $query = "DELETE FROM meddelanden WHERE IDmeddelande='$rensa'";

        $this->query($query);
    }
}
// Function for display messages from the db
function get($meddelande, $msgdate=false)
{
    global $perm, $auth;

    if ($msgdate == true)
    {

```

```

        $query = "SELECT
IDmeddelande,rubrik,meddattid,meddelande,IDpersonal,mbefattning FROM meddelanden
WHERE meddattid >= '."_GAIA_DATE."' ORDER BY meddattid DESC";
    }
    else
    {
        $query = "SELECT IDmeddelande, rubrik, meddattid, meddelande,
IDpersonal, mbefattning FROM meddelanden ORDER BY meddattid DESC";
    }
    // If something goes wrong the sql-query is displayed on the screen
    if ($this->debug)
    {
        echo $query;
    }

    $this->query($query);
    $i = 0;
    $db = new db_gaia;

    // Loops through all messages putting them into an array
    while($this->next_record())
    {
        //htmlspecialchars - convert special characters like tags
        //substr - removes last tree characters from datetime (seconds)
        //stripslashes - Un-quote string
        $this->meddelande[$i][mbefattning] = $this->f(mbefattning);

        $permission = striistr($this->meddelande[$i][mbefattning], $auth-
>auth[IDbefattning]);

        //Control witch messages are shown, depending on what position the
current user has
        if($permission)
        {
            $this->meddelande[$i][IDmeddelande] = $this->f(IDmeddelande);
            $this->meddelande[$i][rubrik] =
htmlspecialchars(stripslashes($this->f(rubrik)));
            $this->meddelande[$i][meddattid] = substr($this->f(meddattid),
0, -3);
            $this->meddelande[$i][meddelande] =
htmlspecialchars(stripslashes($this->f(meddelande)));
            $this->meddelande[$i][IDpersonal] = $this->f(IDpersonal);

            // Fetch Fullname and position from personal and position table
            $db->query("select personal.enamn, personal.fnamn, befattning.befattning
from personal, befattning where personal.IDpersonal='".$this->f(IDpersonal)."'
and befattning.IDbefattning=personal.IDbefattning");
            $db->next_record();
            $this->meddelande[$i][fullname] = $db->f(fnamn).' '.$db->f(enamn);
            $this->meddelande[$i][befatt] = $db->f(befattning);
            $i++;
        }
    }
}
}
?>

```

tis_office.class

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

```

```
// Class with functions for handling offices
// The office class inherits the functions from class db_gaia
class tis_office extends db_gaia
{
    var $debug      = false;

    // Function for build a drop-down-list with all the offices in
    function build_ddlb($IDkontor = false)
    {
        $query = "SELECT * FROM kontor";

        // If something goes wrong the sql-query is displayed on the screen
        if($this->debug)
        {
            echo $query;
        }

        // Function for connecting to db and execute the SQL-query
        $this->query($query);

        // Build the drop-down-list
        echo '<select name="IDkontor" style="color: #CC6600">';
        echo '<option>-Kontor-</option>';

        while($this->next_record())
        {
            if ($IDkontor == $this->f(IDkontor))
            {
                echo '<option value="'. $this->f(IDkontor).'" selected>'. $this->f(kontor). '</option>';
            }
            else
            {
                echo '<option value="'. $this->f(IDkontor).'">'. $this->f(kontor). '</option>';
            }
        }
        echo '</select>';
    }
}
?>
```

tis_position.class

```
<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

class tis_position extends db_gaia
{
    var $debug      = false;

    function get()
    {
        $query = "SELECT * FROM befattning";

        $this->query($query);
    }

    function build_ddlb($IDbefattning = false)
    {
        $query = "SELECT * FROM befattning";
        if($this->debug)
```

```

    {
        echo $query;
    }

    $this->query($query);

    echo '<select name="IDbefattning" style="color: #CC6600">';
    echo '<option>-Befattning-</option>';
    while($this->next_record())
    {
        if ($IDbefattning == $this->f(IDbefattning))
        {
            echo '<option value="'. $this->f(IDbefattning).'" selected>'. $this->f(befattning). '</option>';
        }
        else
        {
            echo '<option value="'. $this->f(IDbefattning).'">'. $this->f(befattning). '</option>';
        }
    }

    echo '</select>';
}
?>

```

tis_user.class

```

<?php
/*This page uses functionality from the framework GAIA.
A lot of ideas come from www.php.net*/

// Class with functions for handling users, like add, remove and update
// The user class inherits the functions from class db_gaia
class tis_user extends db_gaia
{
    var $debug      = false;
    var $personal  = array(array(
                            IDpersonal    => false,
                            fnamn         => false,
                            enamn         => false,
                            losen         => false,
                            epost         => false,
                            nickname      => false,
                            telnr         => false));

    //array for errormessages
    var $error      = array();

    function validate($personal, $IDbefattning, $IDkontor)
    {
        //trim - Removes whitespaces from beginning and end of the string
        //strlen - checks if the variable contains more than 1 character
        if(strlen(trim($personal[fnamn]))<1 || strlen(trim($personal[enamn]))<1 ||
        strlen(trim($personal[nickname]))<1)
        {
            // $this->error[] points to it's own class
            $this->error[] = " Du måste fylla i alla obligatoriska fält ";
        }
        //pass must contain 5 characters
        if(strlen(trim($personal[losen]))<5)
        {

```

```

    $this->error[] = " Lösenordet är felaktigt angivet ";
}
//is_numeric - checks if the value is a numeric value
if (!is_numeric($IDbefattning))
{
    $this->error[] = " Du måste välja befattning ";
}
if (!is_numeric($IDkontor))
{
    $this->error[] = " Du måste välja ett kontor ";
}
//Check if the array contains anything
if(count($this->error)>0)
{
    return false;
}
return true;
}

//Function for adding users to the database
function add($personal, $IDbefattning, $IDkontor)
{
    //addslashes - quote string with slashes
    $personal[fnamn] = addslashes($personal[fnamn]);
    $personal[enamn] = addslashes($personal[enamn]);
    $personal[losen] = addslashes($personal[losen]);
    $personal[epost] = addslashes($personal[epost]);
    $personal[nickname] = addslashes($personal[nickname]);
    $personal[telnr] = addslashes($personal[telnr]);

    //Creates a unique string-value for IDpersonal
    $IDpersonal = md5(uniqid(microtime(),1));

    $query = "INSERT INTO personal (IDpersonal, enamn, fnamn, losen, epost,
nickname, telnr, IDkontor, IDbefattning) VALUES ('$IDpersonal', '$personal[enamn]',
'$personal[fnamn]', '$personal[losen]', '$personal[epost]',
'$personal[nickname]', '$personal[telnr]', '$IDkontor', '$IDbefattning')";

    // Function for connecting to db and execute the SQL-query
    $this->query($query);

    /*When a new user is added and have an e-mail address an autogenerated e-
mail
    will be sent to the new user with nickname and password*/
    $message = "Hej ".$personal[fnamn]."! \n";
    $message.= "Du kan logga in på "._GAIA_URL." \n";
    $message.= "Du har användarnamn: ".$personal[nickname]."\n och lösenord:
".$personal[losen].".";

    mail("$personal[epost]","Välkommen till TIS","$message", "From: Skaraborgs
Städ AB");
    return true;
}
//Removes by administrator choosen user
function remove($IDpersonal)
{
    if(isset($IDpersonal))
    {
        $query = "DELETE FROM personal WHERE IDpersonal='$IDpersonal'";
        $this->query($query);
    }
}
//Updates user information of choosen user
function update($personal, $IDkontor, $IDbefattning)

```



```

{
    // is_array - Checks if the recieved variable is an array
    if(is_array($personal))
    {
        $query = "UPDATE personal SET fnamn='$personal[fnamn]',
enamn='$personal[enamn]', losen='$personal[losen]', epost='$personal[epost]',
nickname='$personal[nickname]', telnr='$personal[telnr]', IDkontor='$IDkontor',
IDbefattning='$IDbefattning' WHERE IDpersonal='$personal[IDpersonal]'";

        $this->query($query);
    }
}

// Gets all users from db into an multi dimentional array
function get($IDpersonal = false)
{
    if (isset($IDpersonal))
    {
        $query = "SELECT * FROM personal WHERE IDpersonal='$IDpersonal'";
    }

    // If something goes wrong the sql-query is displayed on the screen
    if ($this->debug)
    {
        echo $query;
    }

    $this->query($query);
    $i = 0;

    // Loops through all users putting them into an array
    while($this->next_record())
    {
        //htmlentities - convert both double and single quotes
        //stripslashes - Un-quote string
        $this->personal[$i][IDpersonal] = $this->f(IDpersonal);
        $this->personal[$i][fnamn] = htmlentities(stripslashes($this-
>f(fnamn)), ENT_QUOTES);
        $this->personal[$i][enamn] = htmlentities(stripslashes($this-
>f(enamn)), ENT_QUOTES);
        $this->personal[$i][epost] = htmlentities(stripslashes($this-
>f(epost)), ENT_QUOTES);
        $this->personal[$i][losen] = htmlentities(stripslashes($this-
>f(losen)), ENT_QUOTES);
        $this->personal[$i][telnr] = htmlentities(stripslashes($this-
>f(telnr)), ENT_QUOTES);
        $this->personal[$i][nickname] = htmlentities(stripslashes($this-
>f(nickname)), ENT_QUOTES);
        $this->personal[$i][IDbefattning] = $this->f(IDbefattning);
        $this->personal[$i][IDkontor] = $this->f(IDkontor);
        $i++;
    }
}

// Function for building a drop-down-list with all users
function build_ddlb($anvandare = false)
{
    $query = "SELECT IDpersonal,fnamn,enamn FROM personal ORDER BY enamn ASC";

    if($this->debug)
    {
        echo $query;
    }

    $this->query($query);
}

```

```
// Build a drop-down-list with users lastname and firstname
echo '<select name="anvandare" style="color: #CC6600"
onchange="submit()">';
echo '<option>-Användare-</option>';

while($this->next_record())
{
    if ($anvandare == $this->f(IDpersonal))
    {
        echo '<option value="'. $this->f(IDpersonal).'" selected>'. $this-
>f(enamn). ' '. $this->f(fnamn). '</option>';
    }
    else
    {
        echo '<option value="'. $this->f(IDpersonal).'">'. $this->f(enamn). '
'. $this->f(fnamn). '</option>';
    }
}
echo '</select>';
}
?>
```
